PSTAT 126

Regression Analysis

Laura Baracaldo & Rodrigo Targino

Lecture 10 & 11
Transformations and Generalized Least Squares

Violation of Assumptions

We can order model assumptions according to their importance:

1. The systematic form of the model. Wrong assumptions lead to inaccurate predictions.
2. Dependence of errors. We may overlook information about more complex structures.
3. Non-constant variance. Wrong quantification of of prediction uncertainty, however, inference may not be seriously compromised.
4. Normality. For large data sets, the inference will be robust to a lack of normality (due to the Central limit theorem (CLT)), unless the errors are extremely abnormal.

How to overcome Assumptions Violations?

- When linear dependence between the response $\boldsymbol{y}$ and the predictors $\boldsymbol{X}$ is violated we can opt for ***Transformations*** of the response and/or predictors.
- Dependence and heteroscedasticity of the errors can be fixed by changing the covariance matrix structure of the errors - ***Generalized Least Squares (GLS)***.

Transforming the Response

Suppose that the true response-predictors relationship can be written as:

$$y = \exp\left(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p\right) \exp\left(\epsilon\right)$$

- The dependency between the response and the set of predictors is not linear.
- The random error ϵ enters the model *multiplicatively* and not *additively*.

Applying what we have learned on MLR models (Inference/prediction/diagnosis) could be problematic.

Transforming the Response

By taking the log transformarion on both sides of the previous model:

$$log(y) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p + \epsilon$$

We can use results from MLR models on the transformed response $log(y)$.
In practice, we may not know:

- How the errors enter the model.
- What transformation on the response guarantees a linear relationship with the set of predictors. The best approach is to try different transforms, then check the residuals to see whether they satisfy the conditions required for linear regression.

Interpretation and Back Transformation

If we are interested in prediction we may need to obtain those predictions in the original scale, which requires back transforming. For example in the logged model the prediction in the transformed scale is:

$$log(\hat{y}) = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \ldots + \hat{\beta}_p x_p$$

However in the original scale the prediction would be:

$$\hat{y} = \exp(\hat{\beta}_0 + \hat{\beta}_1 x_1 + \ldots + \hat{\beta}_p x_p)$$

Prediction intervals: For a prediction interval in the transformed scale $[l, u]$, the prediction interval in the original scale would be: $[exp(l), exp(u)]$. Note that this interval may not be symmetric.

Interpretation and Back Transformation

Interpretation of Coefficients β_j: Regression coefficients will need to be interpreted with respect to the transformed scale. When you use a log transformation on the response, the regression coefficients have a particular interpretation:

$$\hat{y} = e^{\hat{\beta}_0} e^{\hat{\beta}_1 x_1} \dots e^{\hat{\beta}_p x_p}$$

An increase of one in x_j would multiply the predicted response (in the original scale) by $e^{\hat{\beta}_j}$. Thus when a log scale is used, the regression coefficients can be interpreted in a multiplicative rather than additive manner.

Box-Cox Transformation

The Box-Cox method is a popular tool to determine the most suitable transformation for the response. The method transforms **strictly positive** response $y \to g_\lambda(y)$, where the family of transformations indexed by λ is:

$$g_\lambda(y) = \begin{cases} \frac{y^\lambda - 1}{\lambda} & \lambda \neq 0 \\ \log(y) & \lambda = 0 \end{cases}$$

For $y > 0$, $g_\lambda(y)$ is continuous in λ.

Assumptions:

- $E(g_\lambda(y)) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p = x^T \beta$
- $Var(g_\lambda(y)) = \sigma^2$
- $g_\lambda(y)$ is approximately normal.

Box-Cox Transformation

Reminder: If Z is a transformation of Y, then $p_Y(y) = p_Z(z(y)) \left| \frac{dz}{dy} \right|$.

We estimate λ via maximum likelihood estimation. Under normality of the errors the likelihood is written as:

$$L(\lambda) = (2\pi\sigma^2)^{-\frac{n}{2}} \exp\left\{ -\frac{1}{2\sigma^2} \sum_{i=1}^{n} (g_\lambda(y_i) - x_i^T \beta)^2 \right\} \prod_{i=1}^{n} y_i^{\lambda - 1}$$

Therefore, the log-likelihood is:

$$l(\lambda) = -\frac{n}{2} \log(SSR_\lambda / n) + (\lambda - 1) \sum_{i=1}^{n} \log(y_i)$$

SSR_λ is the SSR when $g_\lambda(y)$ is the response.

Box-Cox Transformation

Transforming the response can make the model harder to interpret so we try to avoid it unless it is necessary. The way we can check this is by using a confidence interval for λ. An approximate $100(1-\alpha)\%$ CI for λ is:

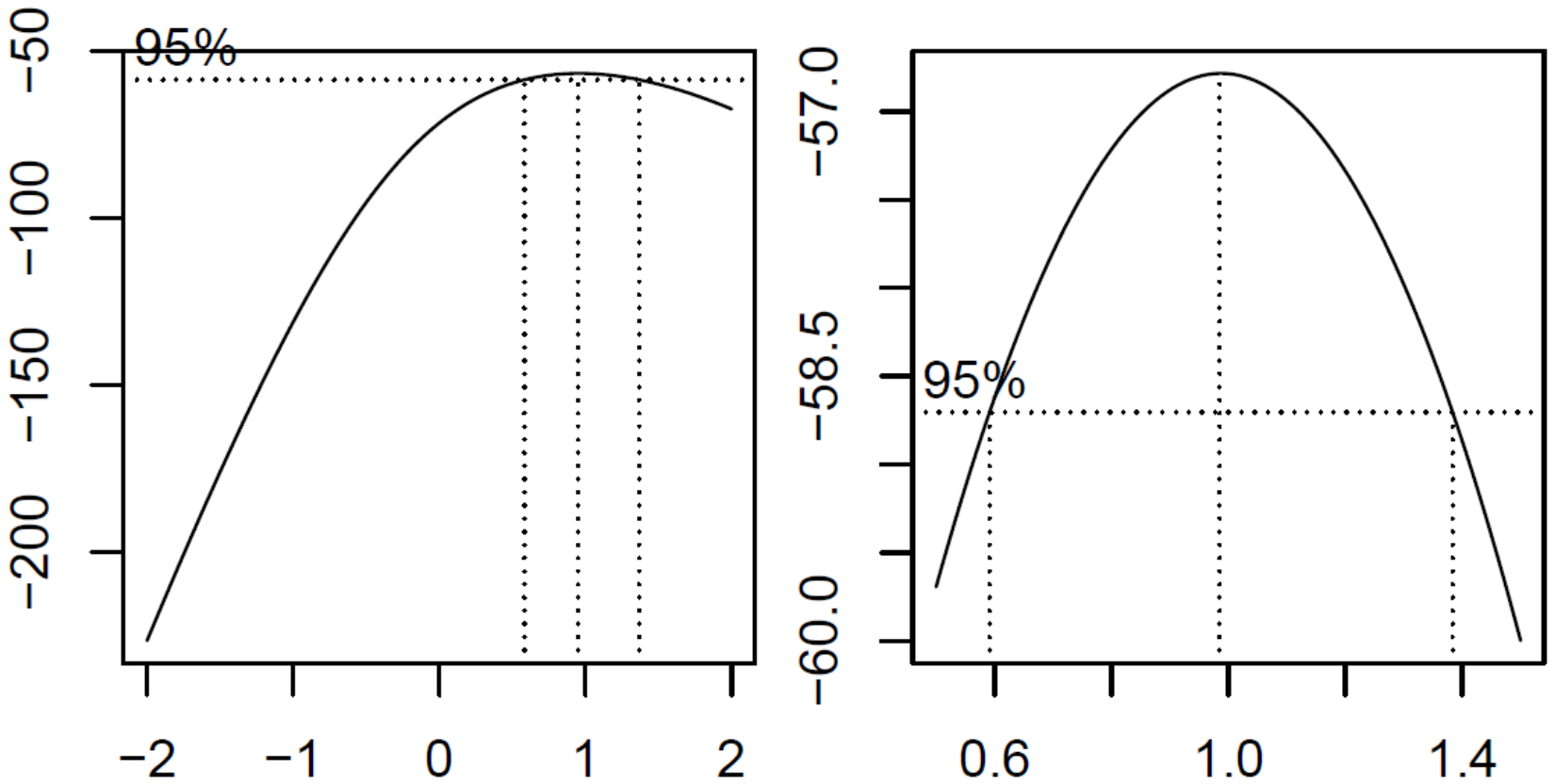
$$\left\{\lambda : l(\lambda) > l(\hat{\lambda}) - \frac{1}{2}\chi^2_{1,1-\alpha}\right\}$$

- If the confidence interval contains $\hat{\lambda} = 1$, no transformation is necessary.
- If the purpose of the regression analysis is interpretation, round off $\hat{\lambda}$. For instance if $\hat{\lambda} = 0.48$ round to the nearest interpretable value $\hat{\lambda} = 0.5$, as $g_{0.5} \propto \sqrt{y}$ is easier to explain.

Box-Cox Transformation

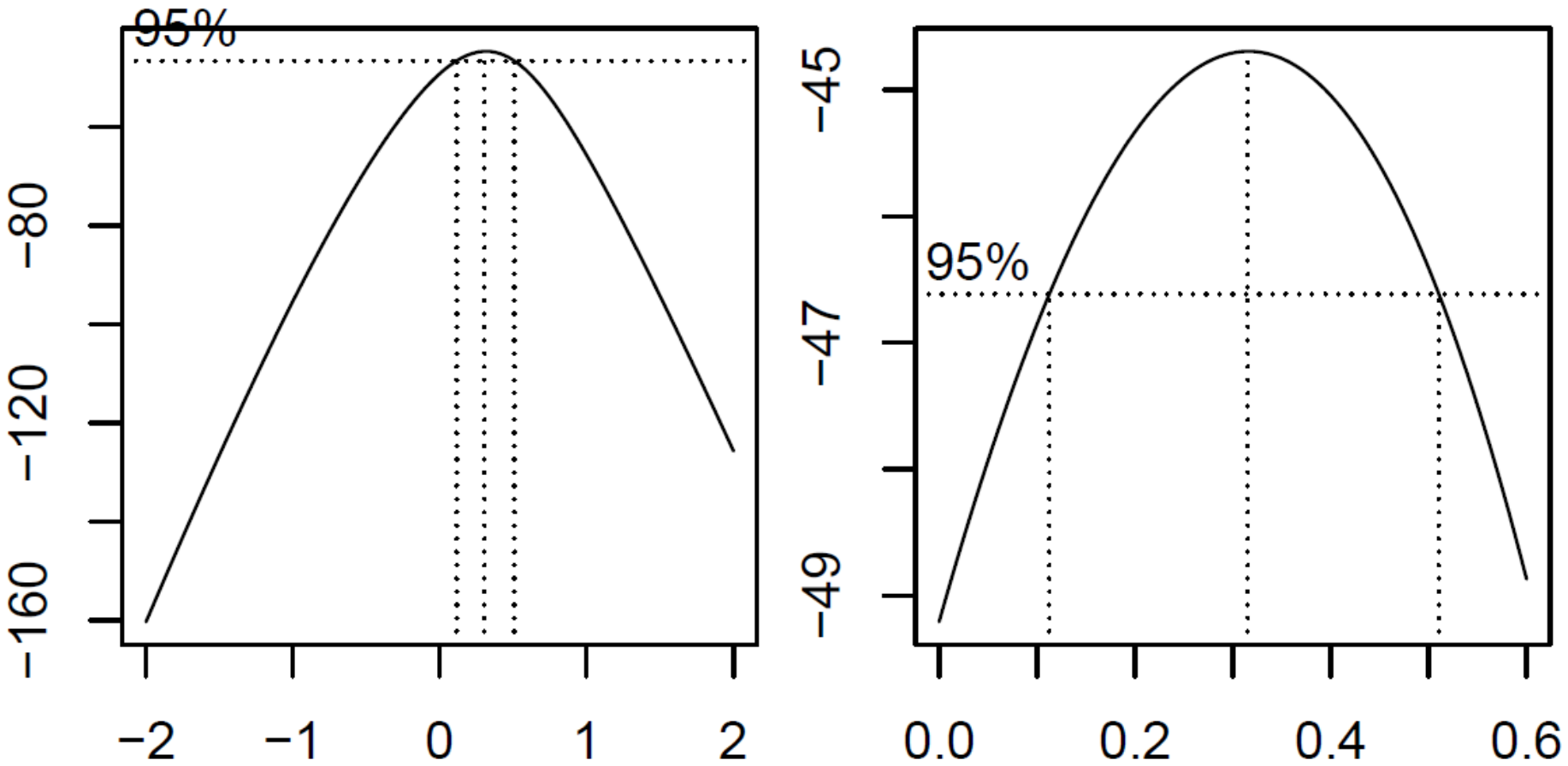
```
## Loading required package: MASS

## Warning: package 'faraway' was built under R version 4.1.3
```

```
lmod<- lm(sr ~ pop15 + pop75+ dpi+ ddpi, savings)
par(mfrow=c(1,2), mar = c(2, 2, 0.8, 0.5))
boxcox(lmod, plotit =TRUE);boxcox(lmod, plotit =TRUE, lambda=seq(0.5, 1.5, by=0.5))
```



Box-Cox Transformation

```
lmod1<- lm(Species ~Area + Elevation+ Nearest+ Scruz+Adjacent, gala)
par(mfrow=c(1,2), mar = c(2, 2, 0.8, 0.5))
boxcox(lmod1, plotit =TRUE);boxcox(lmod1, plotit =TRUE, lambda=seq(0.0, 0.6, by=0.2))
```



General Considerations

1. The Box-Cox method gets greatly impacted by outliers, if you find that $\hat{\lambda} \geq 5$, you may want to check for outliers that justify this extreme transformation.
2. If $y_i < 0$, we can add a constant to all the responses, This is a reasonable solution when the constant added is small.
3. If $\max_i y_i / \min_i y_i$ is small, then the Box-Cox method will not have much real effect.
4. There are other methods for transforming the response (Example: The log family $g_\alpha = \log(y + \alpha)$).

Extensions for Linear Models - Predictors Transformation

Recall that the relationship between the response and the predictors is assumed to be *linear* and *additive*.

- **Polynomial Regression:** One way to generalize the linear structure is by including polynomial terms. In the one-predictor case, we have:

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots + \beta_d x^d + \epsilon$$

Which allows for a more flexible relationship. Note y is no longer linear in x, however it is linear in the z_k's, $z_k = x^k$:

$$y = \beta_0 + \beta_1 z_1 + \beta_2 z_2 + \ldots + \beta_d z_d + \epsilon$$

There are two commons ways to choose d.

1. Keep adding terms until the added term is not statistically significant
2. Start with a large d, remove non-statistically significant terms starting with the highest order term.

Polynomial Regression

```
summary(m1<-lm(sr~ddpi, savings))$coefficients
```

```
##             Estimate Std. Error  t value     Pr(>|t|)
## (Intercept) 7.883021  1.0110011 7.797243 4.464697e-10
## ddpi        0.475830  0.2146166 2.217117 3.138509e-02
```

```
summary(m2<-lm(sr~ddpi + I(ddpi^2), savings))$coefficients #We add quadratic term
```

```
##                Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)  5.13038069 1.43471517  3.575888 0.0008211413
## ddpi         1.75751897 0.53772368  3.268443 0.0020258542
## I(ddpi^2)   -0.09298521 0.03612318 -2.574115 0.0132617330
```

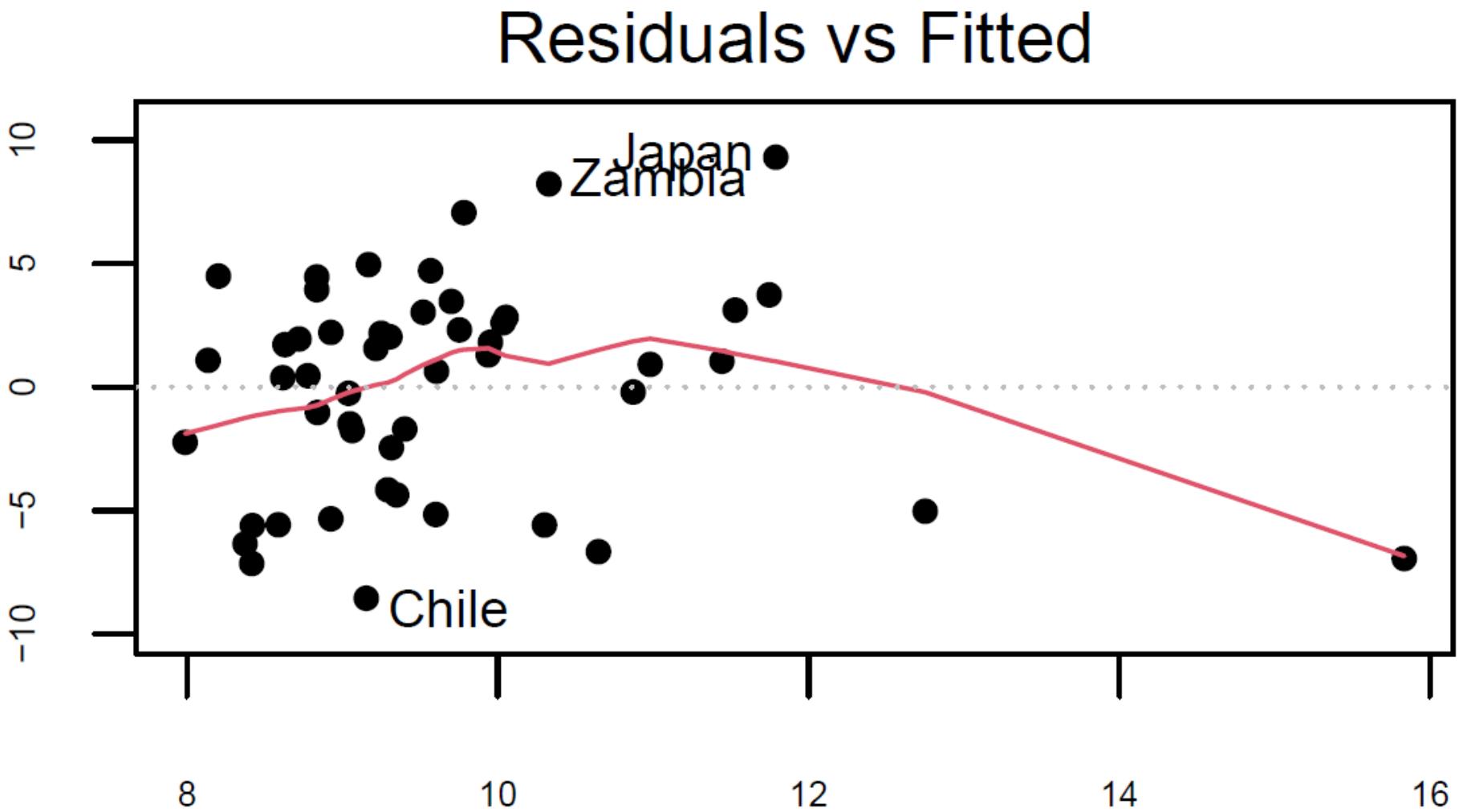
```
summary(m3<-lm(sr~ddpi + I(ddpi^2)+ I(ddpi^3), savings))$coefficients #We add cubic term
```

```
##                  Estimate Std. Error      t value   Pr(>|t|)
## (Intercept)  5.145360e+00 2.19860644  2.340282237 0.02366212
## ddpi         1.746017e+00 1.38045499  1.264812459 0.21230898
## I(ddpi^2)   -9.096724e-02 0.22559835 -0.403226554 0.68864973
## I(ddpi^3)   -8.496955e-05 0.00937393 -0.009064453 0.99280691
```

We stick with the quadratic term.

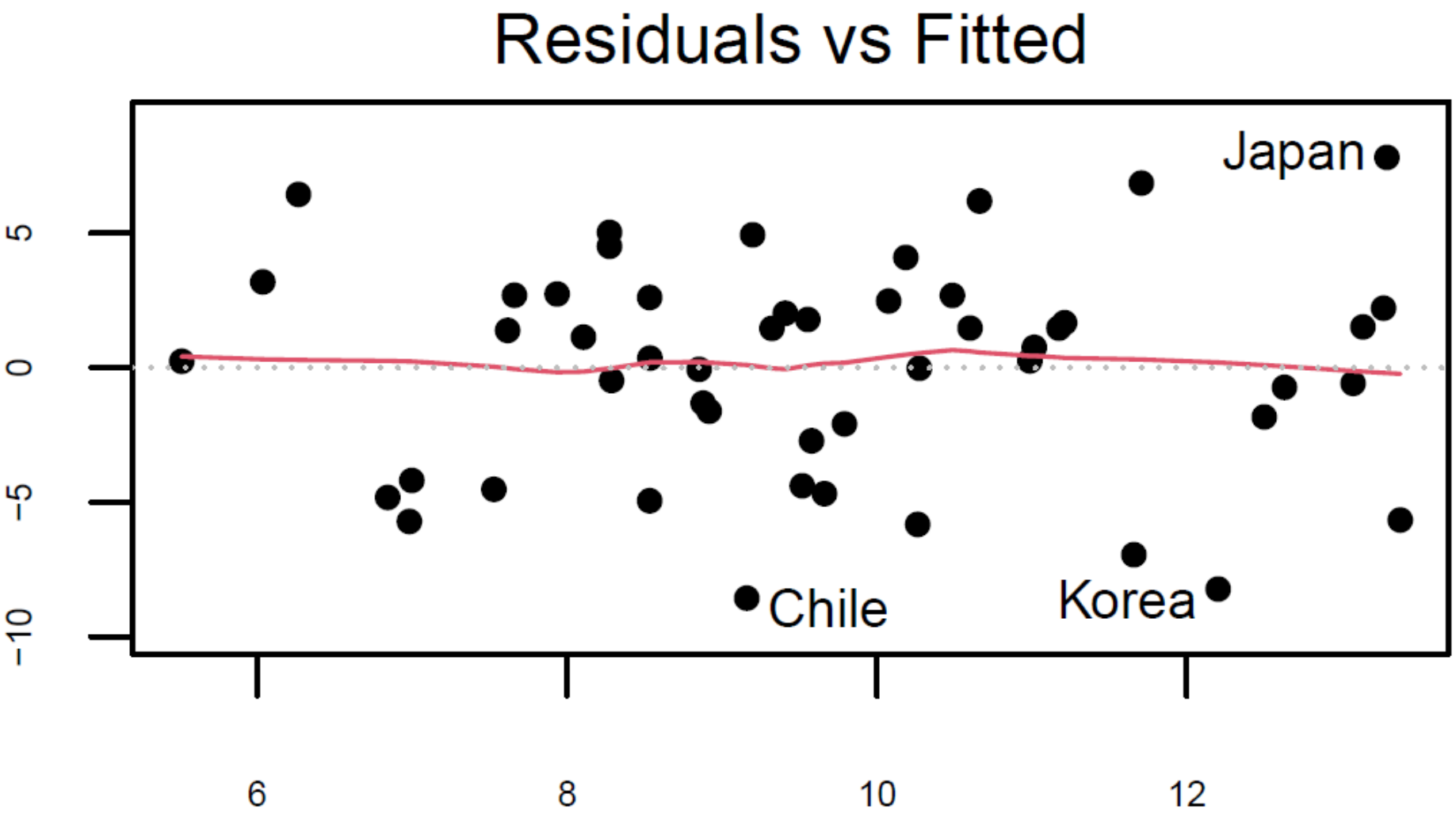
Polynomial Regression

```
par(mar  = c(3, 2, 1.5, 0.5))
plot(m1, cex.main=1, cex.lab=0.5, cex.axis=0.5, pch=20)
```



Polynomial Regression

```
par(mar  = c(3, 2, 1.5, 0.5))
plot(m2, cex.main=1, cex.lab=0.5, cex.axis=0.5, pch=20)
```



Polynomial Regression for $p > 1$

- **Interaction Effects:** We can extend the additive assumption by including non-additive structures through interactions between predictors:

$$y = \beta_0 + \sum_{i=1} \beta_j x_j + \sum_{i \leq j} \alpha_{ij} x_i x_j + \epsilon$$

For $p = 2$:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \alpha_1 x_1^2 + \alpha_2 x_2^2 + \alpha_{12} x_1 x_2 + \epsilon$$

Interactions between x_i and x_j accounts for the impact of x_i on y in the presence of x_j and vice versa.

Polynomial Regression for $p > 1$

```
par(mar  = c(3, 2, 1.5, 0.5))
modelint<- lm(sr ~ pop15*ddpi + I(ddpi^2)
              + I(pop15^2), savings) #By using pop15*ddpi we add degree 1 terms
summary(modelint)$coefficients
```

```
##                Estimate  Std. Error    t value   Pr(>|t|)
## (Intercept) 14.32272231 12.99818146  1.1019020 0.27649541
## pop15       -0.44577097  0.75595872 -0.5896763 0.55842372
## ddpi         2.58494246  1.07001582  2.4157984 0.01991977
## I(ddpi^2)   -0.03567898  0.03972424 -0.8981664 0.37398561
## I(pop15^2)   0.00571870  0.01067209  0.5358557 0.59475693
## pop15:ddpi  -0.04166178  0.03042031 -1.3695381 0.17778077
```

```
modelint2<- lm(sr ~ pop15*ddpi, savings) #Excluding degree 2 terms
summary(modelint2)$coefficients
```

```
##                Estimate Std. Error    t value   Pr(>|t|)
## (Intercept)  7.59867182 4.49133342  1.6918521 0.09743877
## pop15       -0.01520380 0.11373407 -0.1336785 0.89423995
## ddpi         2.59309111 1.05992616  2.4464828 0.01830752
## pop15:ddpi  -0.05464617 0.02651832 -2.0606951 0.04501275
```

Generalized Additive Models

Polynomial Regression is not the only nonlinear prediction transformation. We have also piece wise polynomials and regression splines among others.

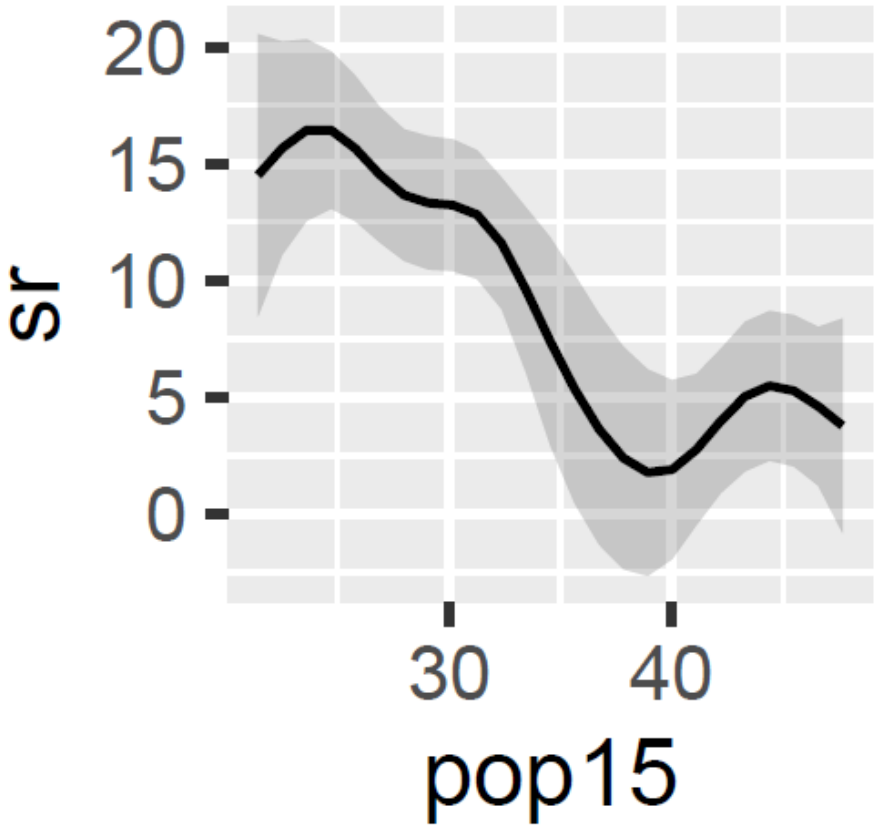
Searching for a good transformation of the predictors is difficult when $p > 1$, since changing the transformation on one predictor may change the choice of transformation for other predictors. We can use the generalized additive models to simultaneously choose the transformation. An additive model takes the form:

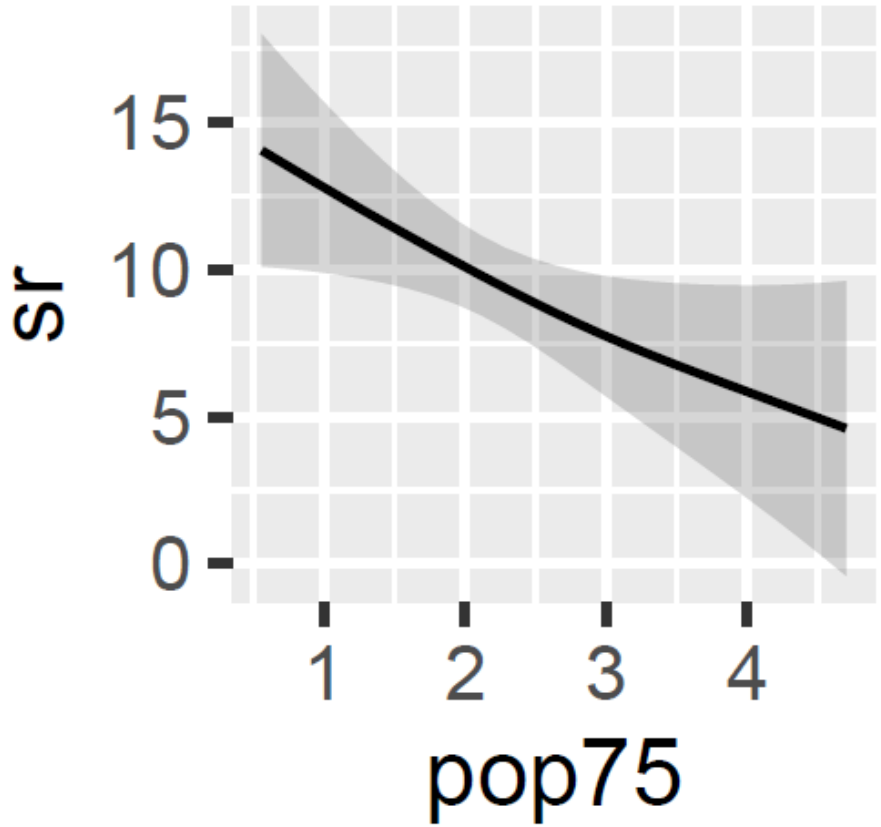
$$y = \alpha + f_1(x_1) + \ldots + f_p(x_p) + \epsilon$$

The linear terms were replaced with more flexible functional forms.

Generalized Additive Models

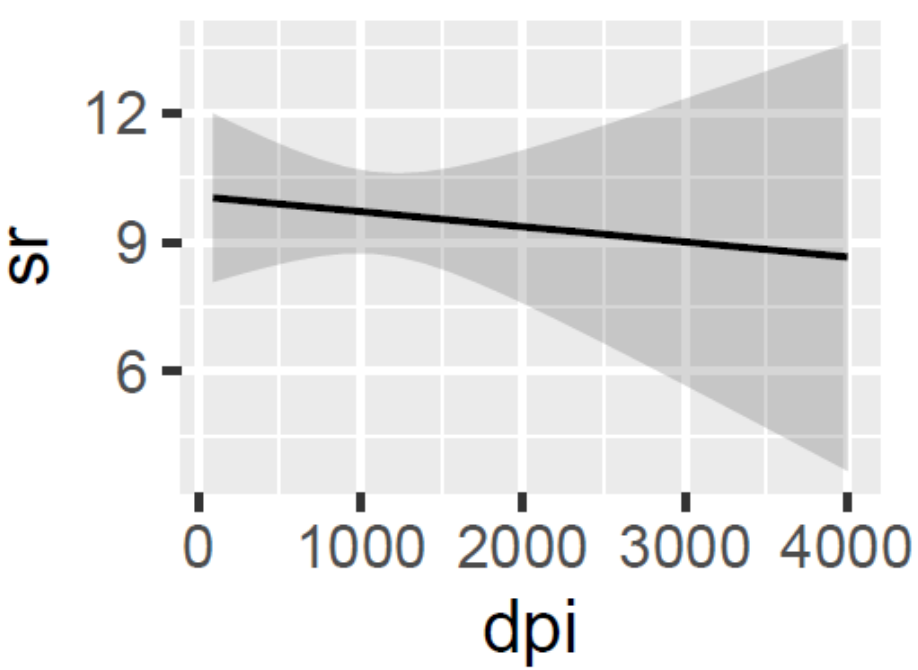
```
require(mgcv); require(ggplot2); require(tidymv);library("gridExtra")
par(mar  = c(3, 2, 0.5, 0.5))
gamod<- gam(sr ~ s(pop15) + s(pop75) + s(dpi) + s(ddpi), data=savings)
p1<-plot_smooths(gamod, pop15); p2<-plot_smooths(gamod, pop75)
grid.arrange(p1, p2,  ncol = 2, nrow = 1)
```

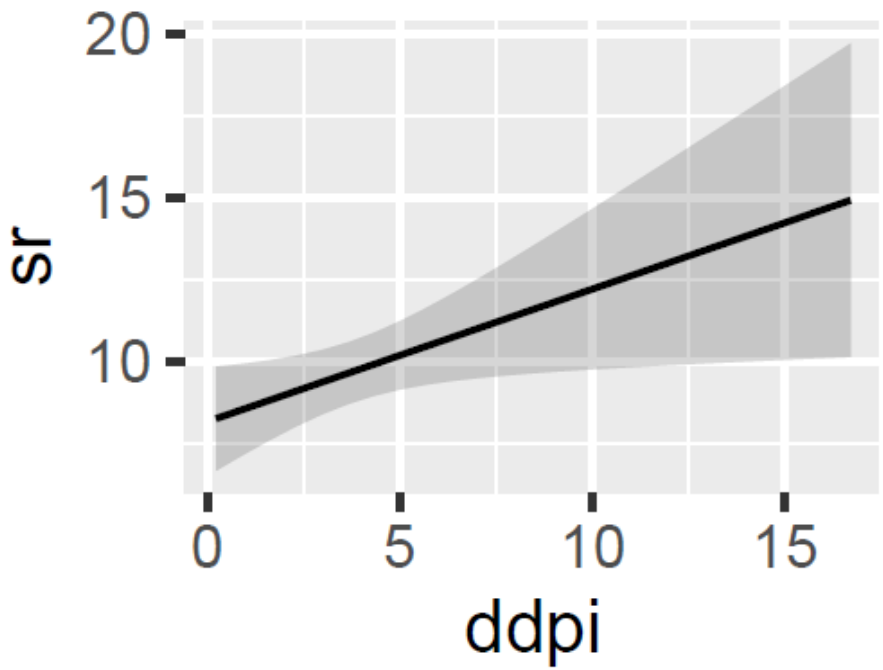




Generalized Additive Models

```
p1<-plot_smooths(gamod, dpi); p2<-plot_smooths(gamod, ddpi)
library("gridExtra")
grid.arrange(p1, p2,  ncol = 2, nrow = 1)
```





Generalized Least Squares

Until now we have assumed that $Var(\boldsymbol{\epsilon}) = Cov(\boldsymbol{\epsilon}) = \sigma^2 \boldsymbol{I}$, which means that the errors have constant variance and are uncorrelated:

$$Cov \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix} = \begin{bmatrix} \sigma^2 & 0 & ... & 0 \\ 0 & \sigma^2 & ... & 0 \\ \vdots & \vdots & ... & \vdots \\ 0 & 0 & ... & \sigma^2 \end{bmatrix}$$

However this may not be the case. A more generalized framework assumes $Var(\boldsymbol{\epsilon}) = Cov(\boldsymbol{\epsilon}) = \sigma^2 \boldsymbol{\Sigma}$ with $\boldsymbol{\Sigma} \neq \boldsymbol{I}$. Although σ^2 is unknown, we assume $\boldsymbol{\Sigma}$ is known.

Generalized Least Squares (GLS)

By using the *Cholesky decomposition*, which can be seen as the "square root" of a matrix, $\boldsymbol{\Sigma}$ can be written as: $\boldsymbol{\Sigma} = \boldsymbol{L}\boldsymbol{L}^T$, with $\boldsymbol{L}$ a lower triangular matrix. We can transform the regression model as follows:

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim N_n(\boldsymbol{0}, \sigma^2\boldsymbol{\Sigma}) \\ \boldsymbol{L}^{-1}\boldsymbol{y} &= \boldsymbol{L}^{-1}\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{L}^{-1}\boldsymbol{\epsilon} \\ \tilde{\boldsymbol{y}} &= \tilde{\boldsymbol{X}}\boldsymbol{\beta} + \tilde{\boldsymbol{\epsilon}} \end{aligned}$$

Note that:
$Var(\tilde{\boldsymbol{\epsilon}}) = Var(\boldsymbol{L}^{-1}\boldsymbol{\epsilon}) = \boldsymbol{L}^{-1}Var(\boldsymbol{\epsilon})(\boldsymbol{L}^{-1})^T = \sigma^2\boldsymbol{L}^{-1}\boldsymbol{L}\boldsymbol{L}^T\boldsymbol{L}^{-T} = \sigma^2\boldsymbol{I}$.
This means we can reduce GLS to ordinary least squares OLS.

Generalized Least Squares (GLS)

As we have transformed the problem to the standard case, we can find estimate for β by minimizing the SSR of the model:

$$\begin{aligned} SSR &= (\tilde{\boldsymbol{y}} - \tilde{\boldsymbol{X}}\boldsymbol{\beta})^T(\tilde{\boldsymbol{y}} - \tilde{\boldsymbol{X}}\boldsymbol{\beta}) \\ &= (\boldsymbol{L}^{-1}\boldsymbol{y} - \boldsymbol{L}^{-1}\boldsymbol{X}\boldsymbol{\beta})^T(\boldsymbol{L}^{-1}\boldsymbol{y} - \boldsymbol{L}^{-1}\boldsymbol{X}\boldsymbol{\beta}) \\ &= (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^T\boldsymbol{L}^{-T}\boldsymbol{L}^{-1}(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) \\ &= (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) \end{aligned}$$

Which is minimized by: $\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{y}$. (Prove this!)

Generalized Least Squares (GLS)

Under normality of the errors it can be proved that:

$$\hat{\boldsymbol{\beta}} \sim N(\boldsymbol{\beta}, \sigma^2(\boldsymbol{X}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{X})^{-1})$$

- *Inference on* $\boldsymbol{\beta}$: Confidence Intervals and Hypothesis Testing are constructed as in previous lectures.
- *Regression Diagnostics*: Since $\tilde{\boldsymbol{\epsilon}} = \boldsymbol{L}^{-1}\boldsymbol{\epsilon} \sim N(\boldsymbol{0}, \sigma^2\boldsymbol{I})$, diagnostics should be applied to the residuals $\tilde{\boldsymbol{\epsilon}}$.
- **Practical Consideration**: The main problem in applying GLS in practice is that $\boldsymbol{\Sigma}$ may not be known and we have to estimate it using the data.

Weighted Least Squares (WLS)

Sometimes is reasonable to assume that the errors are uncorrelated $Cov(\epsilon_i, \epsilon_j) = 0, \quad i \neq j$ with unequal variance $Var(\epsilon_i) = \sigma_i^2$.

$$Cov \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix} = \begin{bmatrix} \sigma_1^2 & 0 & ... & 0 \\ 0 & \sigma_2^2 & ... & 0 \\ \vdots & \vdots & ... & \vdots \\ 0 & 0 & ... & \sigma_n^2 \end{bmatrix} = \boldsymbol{\Sigma}$$

Thus, the Cholesky decomposition of the covariance matrix is written as $\boldsymbol{\Sigma} = \boldsymbol{L}\boldsymbol{L}^T$. With:

$$\boldsymbol{L} = \begin{bmatrix} \sigma_1 & 0 & ... & 0 \\ 0 & \sigma_2 & ... & 0 \\ \vdots & \vdots & ... & \vdots \\ 0 & 0 & ... & \sigma_n \end{bmatrix}$$

Weighted Least Squares (WLS)

We transform the model as in the case of GLS:

$$\boldsymbol{L}^{-1}\boldsymbol{y} = \boldsymbol{L}^{-1}\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{L}^{-1}\boldsymbol{\epsilon}$$
$$\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{X}}\boldsymbol{\beta} + \tilde{\boldsymbol{\epsilon}}$$

Note that $\tilde{\boldsymbol{y}}$ can be seen as a vector of weighted responses:

$$\tilde{\boldsymbol{y}} = \boldsymbol{L}^{-1}\boldsymbol{y} = \begin{bmatrix} \sigma_1^{-1} & 0 & \dots & 0 \\ 0 & \sigma_2^{-1} & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & \sigma_n^{-1} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} y_1/\sigma_1 \\ y_2/\sigma_2 \\ \vdots \\ y_n/\sigma_n \end{bmatrix}$$

Weighted Least Squares (WLS)

We transform the model as in the case of GLS:

$$\boldsymbol{L}^{-1}\boldsymbol{y} = \boldsymbol{L}^{-1}\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{L}^{-1}\boldsymbol{\epsilon}$$
$$\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{X}}\boldsymbol{\beta} + \tilde{\boldsymbol{\epsilon}}$$

And $\tilde{\boldsymbol{X}}$ can be seen as a matrix of weighted predictor variables:

$$\boldsymbol{L}^{-1}\boldsymbol{X} = \begin{bmatrix} 1/\sigma_1 & 0 & \dots & 0 \\ 0 & 1/\sigma_2 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 1/\sigma_n \end{bmatrix} \begin{bmatrix} 1 & x_{11} & \dots & x_{1p} \\ 1 & x_{21} & \dots & x_{2p} \\ \vdots & \vdots & \dots & \vdots \\ 1 & x_{n1} & \dots & x_{np} \end{bmatrix} = \begin{bmatrix} 1/\sigma_1 & \dots & x_{1p}/\sigma_1 \\ 1/\sigma_2 & \dots & x_{2p}/\sigma_2 \\ \vdots & \dots & \vdots \\ 1/\sigma_n & \dots & x_{np}/\sigma_n \end{bmatrix}$$

Weighted Least Squares (WLS)

We transform the model as in the case of GLS:

$$\boldsymbol{L}^{-1}\boldsymbol{y} = \boldsymbol{L}^{-1}\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{L}^{-1}\boldsymbol{\epsilon}$$
$$\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{X}}\boldsymbol{\beta} + \tilde{\boldsymbol{\epsilon}}$$

Similarly, $\tilde{\epsilon}$ can be written as a vector of weighted errors:

$$\tilde{\boldsymbol{\epsilon}} = \boldsymbol{L}^{-1}\boldsymbol{\epsilon} = \begin{bmatrix} \sigma_1^{-1} & 0 & \dots & 0 \\ 0 & \sigma_2^{-1} & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & \sigma_n^{-1} \end{bmatrix} \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix} = \begin{bmatrix} \epsilon_1/\sigma_1 \\ \epsilon_2/\sigma_2 \\ \vdots \\ \epsilon_n/\sigma_n \end{bmatrix}$$

Weighted Least Squares (WLS)

By minimizing the SSR of this model we obtain the Least Squares Estimates:

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^T \boldsymbol{W} \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{W} \boldsymbol{y}$$

Where $\boldsymbol{W}$ is deemed as the weight matrix:

$$\boldsymbol{W} = \begin{bmatrix} W_1 & 0 & ... & 0 \\ 0 & W_2 & ... & 0 \\ \vdots & \vdots & ... & \vdots \\ 0 & 0 & ... & W_n \end{bmatrix} = \begin{bmatrix} 1/\sigma_1 & 0 & ... & 0 \\ 0 & 1/\sigma_2 & ... & 0 \\ \vdots & \vdots & ... & \vdots \\ 0 & 0 & ... & 1/\sigma_n \end{bmatrix}$$

Example: Divorce in the US 1920 – 1996

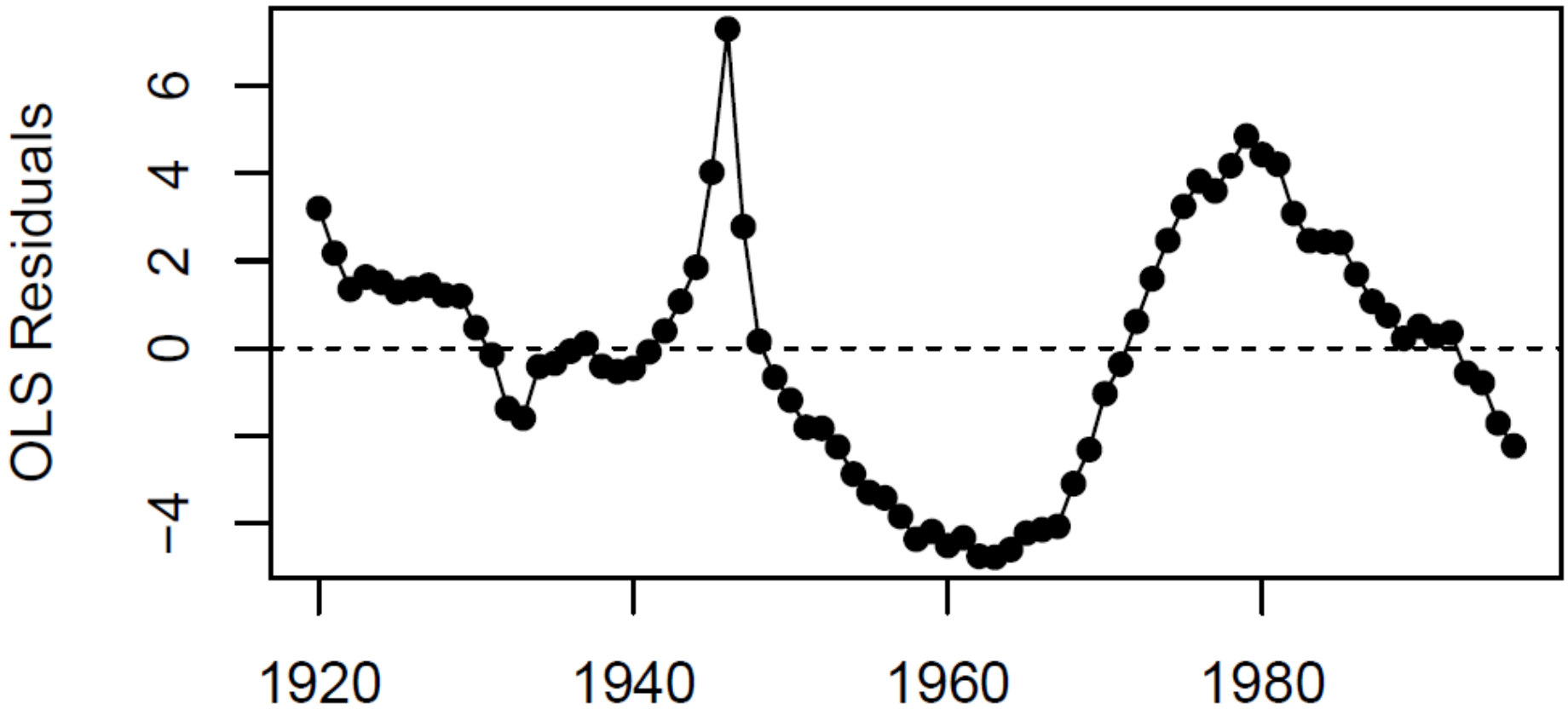
Response: Year (1920-1996), **Predictor**: Divorce (Divorce per women 18 years or more)

```
data(divusa)
lmodt<- lm(divorce ~year, data=divusa)
summary(lmodt)
```

```
##
## Call:
## lm(formula = divorce ~ year, data = divusa)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7828 -1.8092  0.1592  1.6292  7.3048
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -422.97530   27.29465  -15.50   <2e-16 ***
## year           0.22280    0.01394   15.98   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.719 on 75 degrees of freedom
## Multiple R-squared:  0.7731, Adjusted R-squared:   0.77
## F-statistic: 255.5 on 1 and 75 DF,  p-value: < 2.2e-16
```

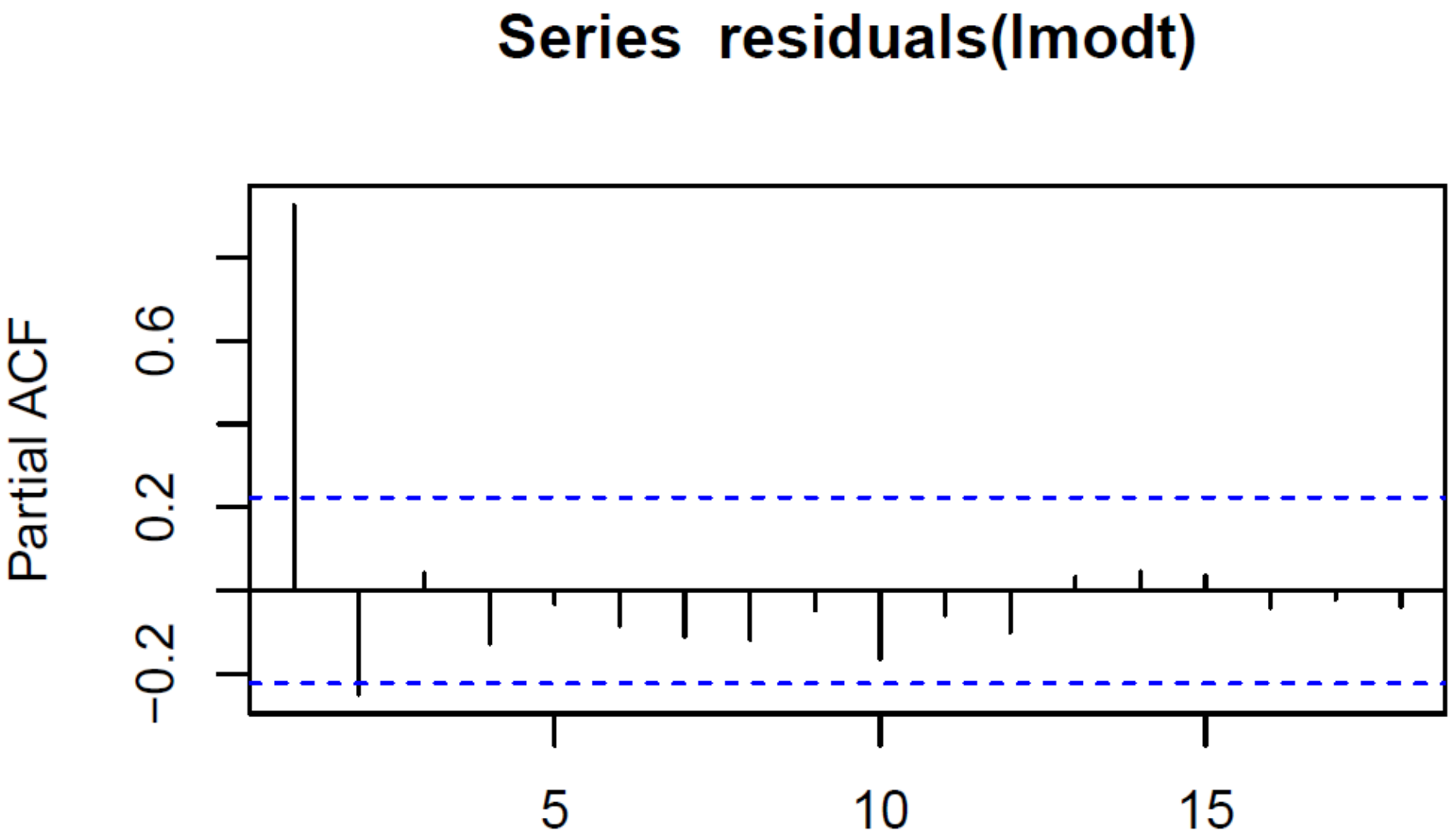
Example: Years vs Residuals

```
plot(divusa$year, residuals(lmodt), type="o", pch=16, xlab="Year", ylab="OLS Residuals")
abline(h=0, lty=2)
```



Example: Autocorrelation

```
acf(residuals(lmodt), type="partial")
```



Example: GLS

```
require(nlme);gmod<- gls(divorce ~year,  correlation=corAR1(form=~year),data=divusa)
summary(gmod)
```

```
## Generalized least squares fit by REML
##   Model: divorce ~ year
##   Data: divusa
##        AIC      BIC    logLik
##   216.2057 225.4756 -104.1028
##
## Correlation Structure: AR(1)
##  Formula: ~year
##  Parameter estimate(s):
##       Phi
## 0.9955474
##
## Coefficients:
##                  Value Std.Error   t-value p-value
## (Intercept) -283.78012 195.07262 -1.454741  0.1499
## year           0.15192   0.09952  1.526571  0.1311
##
##  Correlation:
##      (Intr)
## year -0.999
##
## Standardized residuals:
##        Min         Q1        Med         Q3        Max
## -0.4900705 -0.2675280 -0.1277208  0.2248171  0.6060676
##
## Residual standard error: 9.972029
## Degrees of freedom: 77 total; 75 residual
```