# PE I: Multivariable Regression

## Outliers and Chapter Review
## (Chapter 4)

Andrius Buteikis, andrius.buteikis@mif.vu.lt
http://web.vu.lt/mif/a.buteikis/

# Multiple Regression: Model Assumptions

Much like in the case of the univariate regression with one independent variable, the multiple regression model has a number of required assumptions:

**(MR.1): Linear Model** The Data Generating Process (**DGP**), or in other words, the population, is described by a linear (*in terms of the coefficients*) model:

$$\boldsymbol{Y} = \boldsymbol{X}\beta + \varepsilon \tag{MR.1}$$

**(MR.2): Strict Exogeneity** Conditional expectation of $\varepsilon$, given all observations of the explanatory variable matrix $\boldsymbol{X}$, is zero:

$$\mathbb{E}\left(\varepsilon|\boldsymbol{X}\right) = \mathbf{0} \tag{MR.2}$$

This assumption also implies that $\mathbb{E}(\varepsilon) = \mathbb{E}\left(\mathbb{E}(\varepsilon|\boldsymbol{X})\right) = \mathbf{0}$, $\mathbb{E}(\varepsilon\boldsymbol{X}) = \mathbf{0}$ and $\mathbb{C}\mathrm{ov}(\varepsilon, \boldsymbol{X}) = \mathbf{0}$. Furthermore, this property implies that: $\mathbb{E}\left(\boldsymbol{Y}|\boldsymbol{X}\right) = \boldsymbol{X}\beta$

**(MR.3): Conditional Homoskedasticity** The variance-covariance matrix of the error term, conditional on $\boldsymbol{X}$ is constant:

$$\mathbb{V}\text{ar}\left(\boldsymbol{\varepsilon}|\boldsymbol{X}\right) = \begin{bmatrix} \mathbb{V}\text{ar}(\epsilon_1) & \mathbb{C}\text{ov}(\epsilon_1, \epsilon_2) & ... & \mathbb{C}\text{ov}(\epsilon_1, \epsilon_N) \\ \mathbb{C}\text{ov}(\epsilon_2, \epsilon_1) & \mathbb{V}\text{ar}(\epsilon_2) & ... & \mathbb{C}\text{ov}(\epsilon_2, \epsilon_N) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{C}\text{ov}(\epsilon_N, \epsilon_1) & \mathbb{C}\text{ov}(\epsilon_N, \epsilon_2) & ... & \mathbb{V}\text{ar}(\epsilon_N) \end{bmatrix} = \sigma_\epsilon^2 \boldsymbol{I} \qquad \text{(MR.3)}$$

**(MR.4): Conditionally Uncorrelated Errors** The covariance between different error term pairs, conditional on $\boldsymbol{X}$, is zero:

$$\mathbb{C}\text{ov}\left(\epsilon_i, \epsilon_j|\boldsymbol{X}\right) = 0, \quad i \neq j \qquad \text{(MR.4)}$$

This assumption implies that all error pairs are uncorrelated. For cross-sectional data, this assumption implies that there is no spatial correlation between errors.

**(MR.5)** There exists no exact linear relationship between the explanatory variables. This means that:

$$c_1 X_{i1} + c_2 X_{i2} + ... + c_k X_{ik} = 0, \ \forall i = 1, ..., N \iff c_1 = c_2 = ... = c_k = 0 \quad \text{(MR.5)}$$

This assumption is violated if there exists some $c_j \neq 0$.
Alternatively, this requirement means that:

$$\text{rank}\,(\boldsymbol{X}) = k + 1$$

or, alternatively, that:

$$\det\left(\boldsymbol{X}^\top \boldsymbol{X}\right) \neq 0$$

This assumption is important, because a linear relationship between independent variables means that we cannot separately estimate the effects of changes in each variable separately.

**(MR.6) (optional)** The residuals are normally distributed:

$$\varepsilon | \boldsymbol{X} \sim \mathcal{N}\left(\boldsymbol{0}, \sigma_\epsilon^2 \boldsymbol{I}\right) \quad \text{(MR.6)}$$

# Multiple Regression: Summary of R functions and Theory

# Examining the variables

Note: the code shown is from the example task in chapter 4.11. The code in these slides is only a summarization and therefore not complete.

```r
#From: https://stat.ethz.ch/R-manual/R-devel/library/graphics/html/pairs.html
panel.hist <- function(x, ...){
    usr <- par("usr"); on.exit(par(usr))
    par(usr = c(usr[1:2], 0, 1.5) )
    h <- hist(x, plot = FALSE, breaks = 30)
    breaks <- h$breaks; nB <- length(breaks)
    y <- h$counts; y <- y/max(y)
    rect(breaks[-nB], 0, breaks[-1], y, ...)
}
panel.abs_cor <- function(x, y, digits = 2, prefix = "", cex.cor, ...){
    usr <- par("usr"); on.exit(par(usr))
    par(usr = c(0, 1, 0, 1))
    r <- abs(cor(x, y))
    txt <- format(c(r, 0.123456789), digits = digits)[1]
    txt <- paste0(prefix, txt)
    if(missing(cex.cor)) cex.cor <- 0.8/strwidth(txt)
    text(0.5, 0.5, txt, cex = 2)
}
```

```r
pairs(dt4_train,
      diag.panel = panel.hist,
      lower.panel = panel.smooth,
      upper.panel = panel.abs_cor,
      col = "dodgerblue4",
      pch = 21,
      bg = adjustcolor("dodgerblue3", alpha = 0.2))
```

Correlation matrix can be visualized with:

```
myPanel <- function(x, y, z, ...){
  lattice::panel.levelplot(x,y,z,...)
  my_text <- ifelse(!is.na(z), paste0(round(z, 4)), "")
  lattice::panel.text(x, y, my_text)
}
#
mask = cor(dt4_train)
mask[upper.tri(mask, diag = TRUE)] <- NA
#
#
lattice::levelplot(mask,
                   panel = myPanel,
                   col.regions = viridisLite::viridis(100),
                   main = 'Correlation of numerical variables')
```

## Model estimation

The model $Y_i = \beta_0 + \beta_1 X_{1,i} + \beta_2 X_{2,i} + \beta_3 X_{1,i}^2 + \beta_4 (X_{1,i} \times X_{2,i}) + \epsilon_i$ can be estimated using `lm()`:

```
my_model <- lm(y ~ 1 + x1 + x2 + I(x1^2) + x1*x2, data = dt4_train)
```

Note that

- `y ~ 1 + x1 + x2 + I(x1^2) + x1*x2` is the same as `y ~ 1 + x1 + x2 + I(x1^2) + x1:x2`
- `y ~ 1 + x1 + x2 + I(x1^2) + x1*x2` is the same as `y ~ 1 + x1*x2 + I(x1^2)`
- `y ~ 1 + I(x1^2) + x1:x2` is $Y_i = \beta_0 + \beta_1 X_{1,i}^2 + \beta_2 (X_{1,i} \times X_{2,i}) + \epsilon_i$

Furthermore:

- When specifying polynomial and interaction terms it is important to think about their interpretation;
- In general, the coefficient signs should make economic sense, so you should have an initial assumption of their signs. Then, from the estimated model, you should determine whether the coefficients are significant, and if they are, whether their signs make sense (sometimes both a positive and a negative sign can have an economic interpretation).
- As per the lecture note example, you should be able to write down the estimated regression model.

## Linear Restrictions

$$\begin{cases} H_0 & : \beta_{educ} = \beta_{exper}, \text{ and } \beta_{educ^2} = \beta_{exper^2} \\ H_1 & : \beta_{educ} \neq \beta_{exper} \text{ or } \beta_{educ^2} \neq \beta_{exper^2} \text{ or both} \end{cases}$$

```
car::linearHypothesis(mdl_4_fit, c("educ-exper=0", "I(educ^2)-I(exper^2)=0"))
```

$$\begin{cases} H_0 & : \beta_{educ} = \beta_{exper} \\ H_1 & : \beta_{educ} \neq \beta_{exper} \end{cases}$$

```
car::linearHypothesis(mdl_4_fit, c("educ-exper=0"))
```

## Restricted Least Squares

If we have no grounds to reject the linear restriction hypothesis - we can estimate the model via *RLS*:

```
lrmest::rls(formula = formula(my_model),
            R = LL,
            r = rr,
            data = dt4_train,
            delt = rep(0, length(rr)))
```

Note that the standard errors and $p$-values for the coefficient significance are available provided in the output but can be calculated manually.

## Collinearity

Verify whether the explanatory variables are collinear using the Variance Inflation Factor (VIF):

```
car::vif(my_model)
```

Note that in case of categorical/factor variables, the Generalized VIF will be calculated.

A $\left(\text{GVIF}^{1/(2\cdot \text{DF})}\right)^2 < 5$ (DF is the number of coefficients, polynomial and interaction terms that in some way include the specific variable) is equivalent to a VIF $< 5$, which indicated no multicollinearity.

Carrying out the test on a model without any interaction and polynomial terms. If some variables are found to be collinear- they should be excluded from the model.

## Residual diagnostics - plots

- The residual vs fitted plot:

```
plot(my_model$fitted.values, my_model$residuals)
```

- The residual histogram:

```
hist(my_model$residuals)
```

- The QQ plot:

```
qqnorm(my_model$residuals)
qqline(my_model$residuals, col = "red")
```

## Residual diagnostics - tests

## Homoskedasticity tests

$$\begin{cases} H_0 & : \text{ residuals are homoskedastic} \\ H_1 & : \text{ residuals are heteroskedastic} \end{cases}$$

```
# Breusch-Pagan Test
print(lmtest::bptest(my_model))
# Goldfeld-Quandt Test
lmtest::gqtest(my_model, alternative = "two.sided")
# White Test
lmtest::bptest(mdl_3_fit, ~ x1*x2 + I(x1^2) + I(x2^2), data = dt4_train)
```

In general, if the $p$-value $< 0.05$ - we reject the null hypothesis and conclude that the residuals are heteroskedastic.

## Autocorrelation Tests

$$\begin{cases} H_0 & \text{: the errors are serially uncorrelated} \\ H_1 & \text{: the errors are autocorrelated (the exact order of the autocorrelation depends on the test carried out)} \end{cases}$$

```
# Durbin-Watson Test - first order autocorrelation only
lmtest::dwtest(my_model, alternative = "two.sided")
# Breusch-Godfrey Test
lmtest::bgtest(my_model, order = 2)
```

In general, if the $p$-value $< 0.05$ - we reject the null hypothesis and conclude that the residuals are autocorrelated (or simply, serially correlated).

## Normality Tests

$$\begin{cases} H_0 & \text{: residuals follow a normal distribution} \\ H_1 & \text{: residuals do not follow a normal distribution} \end{cases}$$

```
norm_tests = c("Anderson-Darling", "Shapiro-Wilk", "Kolmogorov-Smirnov",
               "Cramer-von Mises", "Jarque-Bera")
norm_test <- data.frame(
    p_value = c(nortest::ad.test(my_model$residuals)$p.value,
                shapiro.test(my_model$residuals)$p.value,
                ks.test(my_model$residuals, y = "pnorm", alternative = "two.sided")$p.value,
                nortest::cvm.test(my_model$residuals)$p.value,
                tseries::jarque.bera.test(my_model$residuals)$p.value),
    Test = norm_tests)
```

In general, if the $p$-value $< 0.05$ - we reject the null hypothesis and conclude that the residuals are not normally distributed.

- What can you say about the residual plots - are there any non-linearities? Do the residuals appear to be normally distributed?
- What can you conclude about your model from these tests? Which of the **(MR.1)** - **(MR.6)** assumptions are (not) violated?

## HCE

If we find that our residuals are heteroskedastic but not autocorrelated - we can correct the standard errors via either `HC0`, `HC1`, `HC2`, or `HC3`.
Of the four, `HC3` is the superior estimate.

```
lmtest::coeftest(my_model,
                 vcov. = sandwich::vcovHC(my_model, type = "HC3"))
```

## WLS

Alternatively, we can correct the estimates themselves for heteroskedasticity by using WLS with a generic weight function:

```
log_resid_sq_ols <- lm.fit(y = log(my_model$residuals^2), x = model.matrix(my_model))
h_est = exp(log_resid_sq_ols$fitted.values)
#
my_model_wls <- lm(formula = formula(my_model), data = dt4_train, weights =  1 / h_est)
```

Note that $R^2_{adj}$ calculated for *WLS* is not comparable to the OLS $R^2_{adj}$.

## HAC

If the residuals are autocorrelated (and also heteroskedastic, but not necessarily) - we can correct the standard errors via:

```
mtest::coeftest(my_model,
                sandwich::NeweyWest(my_model, lag = 2))[, ], 4)
```

## Model Specification test

### Rainbow Test for Linearity:

```
lmtest::raintest(formula(my_model), order.by = ~ x1, data = dt4_train)
lmtest::raintest(formula(my_model), order.by = ~ x2, data = dt4_train)
...
```

The data needs to be ordered. If $p$-value $< 0.05$ - we reject the null hypothesis and conclude that the model fit is not adequate.

### Ramsey Regression Specification Error Test:

```
lmtest::resettest(formula(my_model), data = dt4_train, power = 2, type = "fitted")
lmtest::resettest(formula(my_model), data = dt4_train, power = 3, type = "fitted")
...
```

If $p$-value $< 0.05$, we reject the null hypothesis and conclude that the original model is inadequate.

## Automatic model selection

We can pass different variables, their interaction and polynomial terms via `my_full_formula` and attempt to automatically fit the best model based on $BIC$, or $R^2_{adj}$ up to a maximum of 8 explanatory variables:

```
leaps::regsubsets(my_full_formula, data = dt4_train, nvmax = 8, nbest = 1)
```

Note: check the lecture notes and the example task chapter on the various plots that are available.