Applied Econometrics with R

Chapter 4

Diagnostics and Alternative Methods of Regression

Diagnostics and Alternative Methods of Regression

Overview

Diagnostics and alternative methods of regression

Validate linear regression models:

- **Regression diagnostics:** Comparison of statistics for full data set and for data with single observations deleted.
- **Diagnostic tests:** Test for heteroskedasticity, autocorrelation, and misspecification of the functional form, etc.
- **Robust covariances:** Covariance estimators that are consistent for a wide class of disturbance structures.

Alternative methods of regression:

- **Resistant regression:** Regression techniques that are robust/resistant to outliers and unusual observations.
- **Quantile regression:** Model quantiles of the conditional distribution of a variable (instead of the conditional mean).

Diagnostics and Alternative Methods of Regression

Regression Diagnostics

Regression diagnostics

Goal: Find points that are not fitted as well as they should be or have undue influence on the fitting of the model.

Techniques: Based on deletion of observations, see Belsley, Kuh, and Welsch (1980).

Illustration: `PublicSchools` data provide per capita `Expenditure` on public schools and per capita `Income` by state for the 50 states of the USA plus Washington, DC., for 1979.

```
R> data("PublicSchools", package = "sandwich")
R> summary(PublicSchools)
```

```
  Expenditure        Income
 Min.   :259   Min.   : 5736
 1st Qu.:315   1st Qu.: 6670
 Median :354   Median : 7597
 Mean   :373   Mean   : 7608
 3rd Qu.:426   3rd Qu.: 8286
 Max.   :821   Max.   :10851
 NA's   :1
```

Regression diagnostics

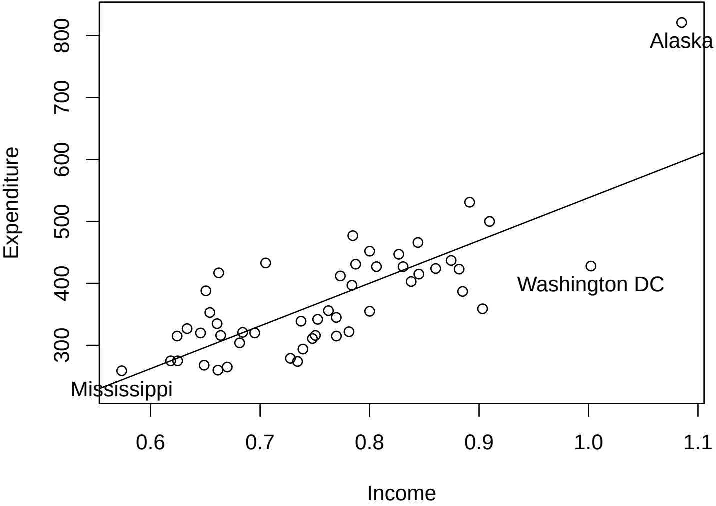
Preprocessing:

- Omit incomplete observations (only Wisconsin) using `na.omit()`.
- Scale income to be in 10,000 USD.

Visualization: Scatterplot with fitted linear model and three highlighted observations.

```
R> ps <- na.omit(PublicSchools)
R> ps$Income <- ps$Income / 10000
R> plot(Expenditure ~ Income, data = ps, ylim = c(230, 830))
R> ps_lm <- lm(Expenditure ~ Income, data = ps)
R> abline(ps_lm)
R> id <- c(2, 24, 48)
R> text(ps[id, 2:1], rownames(ps)[id], pos = 1, xpd = TRUE)
```

Regression diagnostics



Regression diagnostics

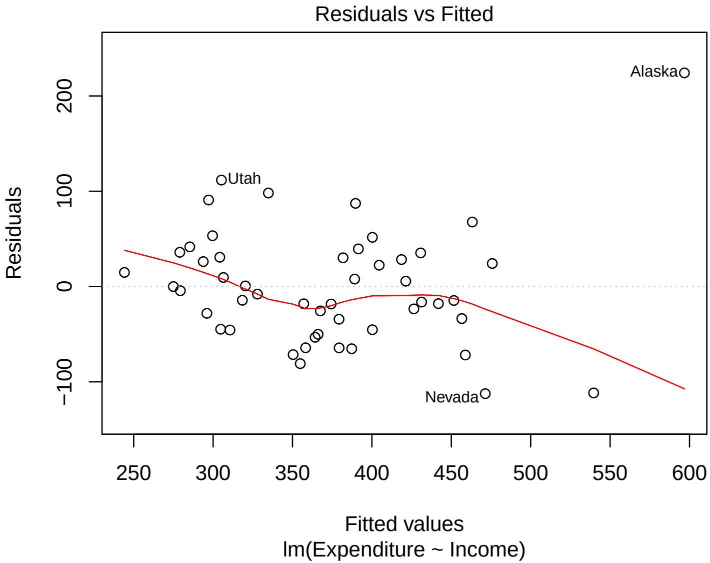
Diagnostic plots: `plot()` method for “`lm`” objects provides

- Residuals versus fitted values (for checking $\mathsf{E}(\varepsilon|X) = 0$).
- QQ plot: ordered residuals versus normal quantiles (for checking normality).
- Scale-location plot: $\sqrt{|\hat{r}_i|}$ (of standardized residuals r_i) versus fitted values $\hat{y}_i$ (for checking i.i.d. assumption, in particular $\mathsf{Var}(\varepsilon|X) = \sigma^2 I$).
- Combinations of standardized residuals, leverage, and Cook’s distance.

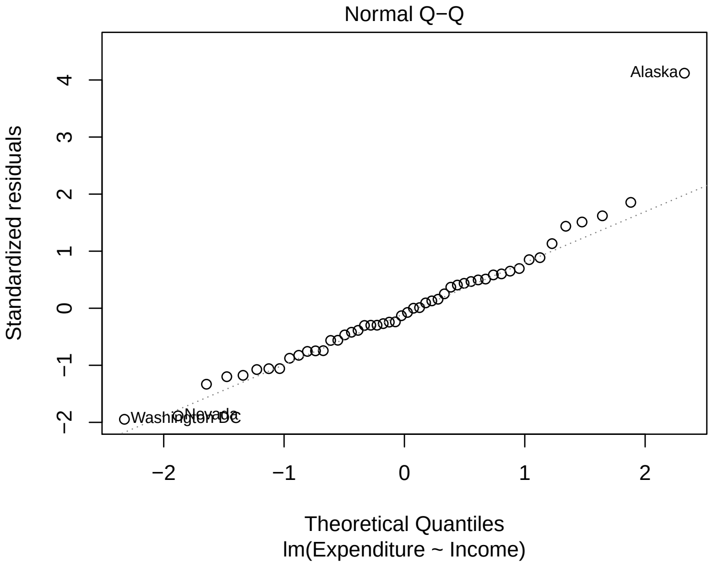
```
R> plot(ps_lm, which = 1:6)
```

By default only four of the six available plots are shown: `which = c(1:3, 5)`.

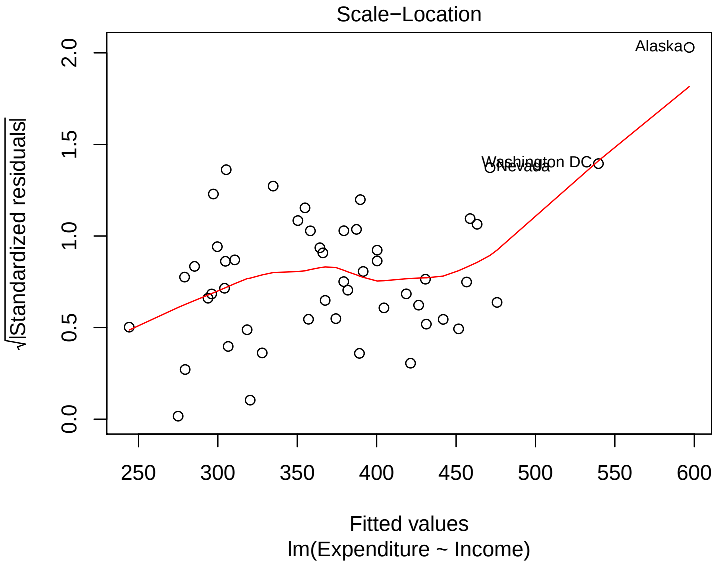
Regression diagnostics



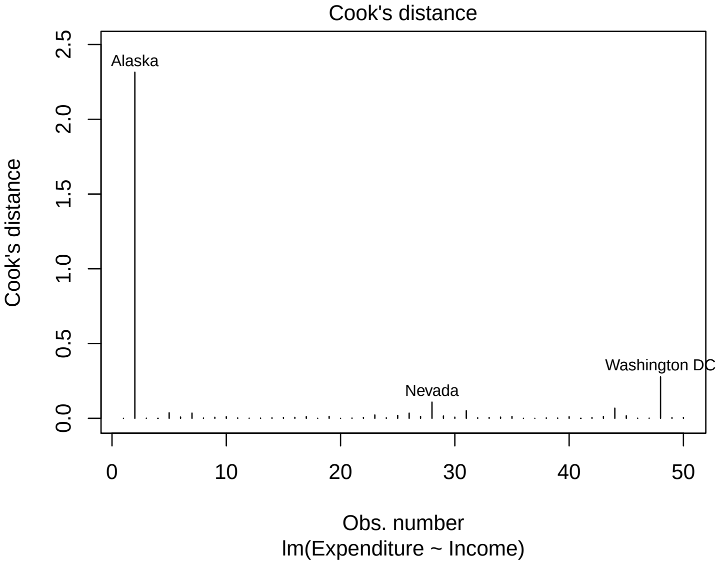
Regression diagnostics



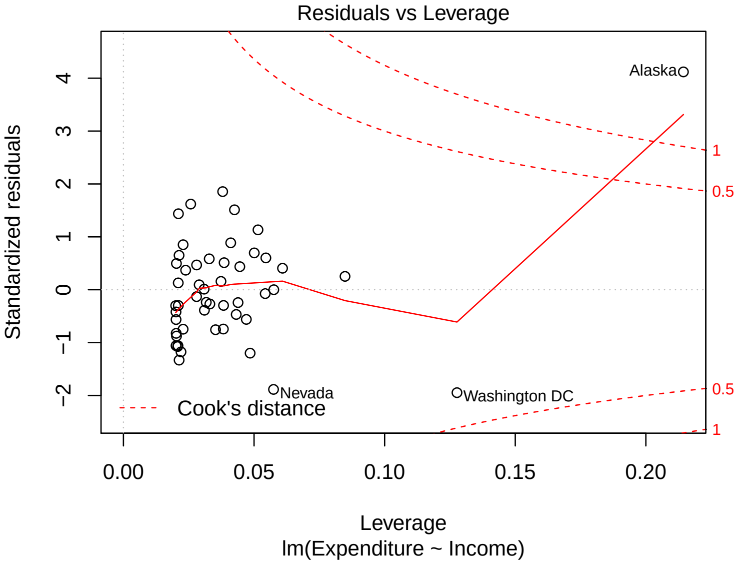
Regression diagnostics



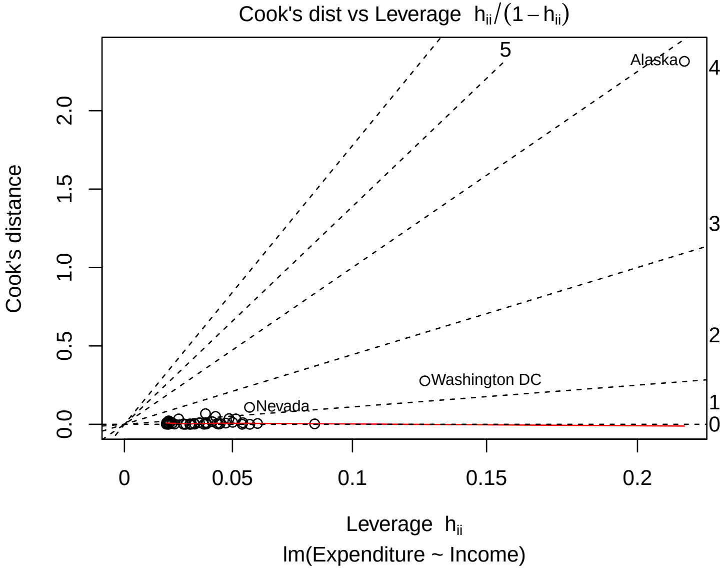
Regression diagnostics



Regression diagnostics



Regression diagnostics



Regression diagnostics

Interpretation: Alaska stands out in all plots.

- Large residual (`which = 1`).
- Upper tail of empirical distribution of residuals (`which = 2`).
- Casts doubt on the assumption of homogeneous variances (`which = 3`).
- Corresponds to an extraordinarily large Cook's distance (`which = 4` and `6`).
- Has the highest leverage (`which = 5` and `6`).

There are further observations singled out, but none of these are as dominant as Alaska.

Leverage and standardized residuals

Recall: OLS residuals are not independent and do not have the same variance.

More precisely: If $\mathrm{Var}(\varepsilon|X) = \sigma^2 I$, then $\mathrm{Var}(\hat{\varepsilon}|X) = \sigma^2(I - H)$.
$H = X(X^\top X)^{-1}X^\top$ is the "hat matrix".

Hat values:

- Diagonal elements h_{ii} of H.
- Provided by generic function `hatvalues()`.
- Since $\mathrm{Var}(\hat{\varepsilon}_i|X) = \sigma^2(1 - h_{ii})$, observations with large h_{ii} will have small values of $\mathrm{Var}(\hat{\varepsilon}_i|X)$, and hence tend to have residuals $\hat{\varepsilon}_i$ close to zero.
- h_{ii} measures the *leverage* of observation i.

Leverage and standardized residuals

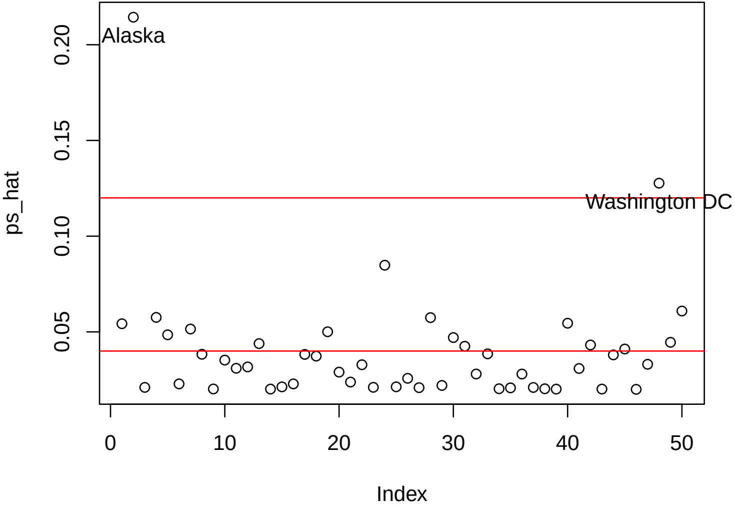
High leverage:

- The trace of H is k (the number of regressors).
- High leverage h_{ii} typically means two or three times larger than average hat value k/n.
- Leverage only depends on X and not on y.
- Good/bad leverage points: high leverage points with typical/unusual y_i.

Visualize hat values with mean and three times the mean:

```
R> ps_hat <- hatvalues(ps_lm)
R> plot(ps_hat)
R> abline(h = c(1, 3) * mean(ps_hat), col = 2)
R> id <- which(ps_hat > 3 * mean(ps_hat))
R> text(id, ps_hat[id], rownames(ps)[id], pos = 1, xpd = TRUE)
```

Leverage and standardized residuals



Leverage and standardized residuals

Standardized residuals: $\text{Var}(\hat{\varepsilon}_i | X) = \sigma^2(1 - h_{ii})$ suggests

$$r_i = \frac{\hat{\varepsilon}_i}{\hat{\sigma}\sqrt{1 - h_{ii}}}.$$

- Sometimes referred to as “internally studentized residuals”.
- Warning: not to be confused with (externally) studentized residuals (defined below).
- If model assumptions are correct: $\text{Var}(r_i | X) = 1$ and $\text{Cor}(r_i, r_j | X)$ tends to be small.
- In R: `rstandard()`.

Deletion diagnostics

Idea: Detection of unusual observations via leave-one-out (or deletion) diagnostics (see Belsley, Kuh, and Welsch 1980).

Notation: Exclude point *i* and compute

- estimates $\hat{\beta}_{(i)}$ and $\hat{\sigma}_{(i)}$,
- associated predictions $\hat{y}_{(i)} = X\hat{\beta}_{(i)}$.

Influential observations: Observations whose removal causes a large change in the fit.

Influential observations may or may not have large leverage and may or may not be an outlier. But they tend to have at least one of these properties.

Deletion diagnostics

Basic quantities:

$$\begin{aligned}
\mathit{DFFIT}_i &= y_i - \hat{y}_{i,(i)}, \\
\mathit{DFBETA} &= \hat{\beta} - \hat{\beta}_{(i)}, \\
\mathit{COVRATIO}_i &= \frac{\det(\hat{\sigma}^2_{(i)}(X_{(i)}^\top X_{(i)})^{-1})}{\det(\hat{\sigma}^2(X^\top X)^{-1})}, \\
D_i^2 &= \frac{(\hat{y} - \hat{y}_{(i)})^\top (\hat{y} - \hat{y}_{(i)})}{k\hat{\sigma}^2}.
\end{aligned}$$

Interpretation:

- *DFFIT*: change in fitted values (scaled version: *DFFITS*).
- *DFBETA*: changes in coefficients (scaled version: *DFBETAS*).
- *COVRATIO*: change in covariance matrix.
- D^2 (Cook's distance): reduce information to a single value per observation.

Deletion diagnostics

In R:

- `dffit()`, `dffits()`,
- `dfbeta()`, `dfbetas()`,
- `covratio()`,
- `cooks.distance()`.

Additionally: `rstudent()` provides the (externally) studentized residuals

$$t_i = \frac{\hat{\varepsilon}_i}{\hat{\sigma}_{(i)}\sqrt{1 - h_{ii}.}}$$

The function `influence.measures()`

Convenience function: Computes most important quantities `dfbetas()`, `dffits()`, `covratio()`, `cooks.distance()`, `hatvalues`.

Influential observations: Observations that are unusual for at least one of the influence measures are highlighted.

```
R> summary(influence.measures(ps_lm))

Potentially influential observations of
	 lm(formula = Expenditure ~ Income, data = ps) :

              dfb.1_  dfb.Incm dffit   cov.r   cook.d  hat
Alaska        -2.39_*  2.52_*   2.65_*  0.55_*  2.31_*  0.21_*
Mississippi    0.07   -0.07     0.08    1.14_*  0.00    0.08
Washington DC  0.66   -0.71    -0.77_*  1.01    0.28    0.13_*
```

The function `influence.measures()`

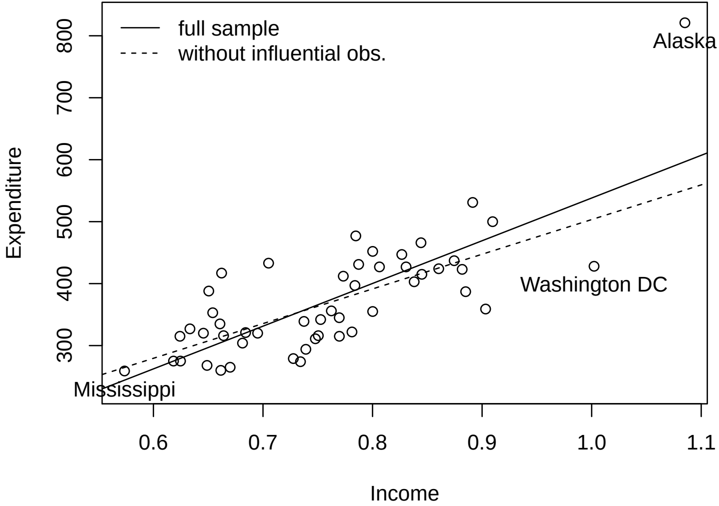
Interpretation:

- Alaska stands out by any measure of influence and is clearly a bad leverage point.
- Washington, DC, seems to be a bad leverage point (but not nearly as bad as Alaska).
- Mississippi is associated with a large change in the covariances.

Exclude influential observations:

```
R> plot(Expenditure ~ Income, data = ps, ylim = c(230, 830))
R> abline(ps_lm)
R> id <- which(apply(influence.measures(ps_lm)$is.inf, 1, any))
R> text(ps[id, 2:1], rownames(ps)[id], pos = 1, xpd = TRUE)
R> ps_noinf <- lm(Expenditure ~ Income, data = ps[-id,])
R> abline(ps_noinf, lty = 2)
R> legend("topleft", c("full sample", "without influential obs."),
+    lty = 1:2, bty = "n")
```

The function `influence.measures()`



Diagnostics and Alternative Methods of Regression

Diagnostic Tests

Diagnostics tests

More formal validation: Diagnostic testing, e.g., for heteroskedasticity in cross-section regressions or disturbance autocorrelation in time series regressions.

In R: Package **lmtest** provides a large collection of diagnostic tests.

Typically, the tests return an object of class "`htest`" (hypothesis test) with test statistic, corresponding *p* value, and additional parameters such as degrees of freedom (where appropriate), the name of the tested model, or the method used.

Background: Underlying theory is provided in Baltagi (2002), Davidson and MacKinnon (2004), and Greene (2003).

Diagnostics tests

Illustration: Reconsider `Journals` data as an example for cross-section regressions.

Preprocessing: As before and additionally include `age` of the journals (for the year 2000, when the data were collected).

```
R> data("Journals", package = "AER")
R> journals <- Journals[, c("subs", "price")]
R> journals$citeprice <- Journals$price/Journals$citations
R> journals$age <- 2000 - Journals$foundingyear
```

Regression: log-subscriptions explained by log-price per citation.

```
R> jour_lm <- lm(log(subs) ~ log(citeprice), data = journals)
```

Testing for heteroskedasticity

Assumption of homoskedasticity $\text{Var}(\varepsilon_i|x_i) = \sigma^2$ must be checked in cross-section regressions.

Breusch-Pagan test:

- Fit linear regression model to the squared residuals $\hat{\varepsilon}_i^2$.
- Reject if too much of the variance is explained by additional explanatory variables, e.g.,
  - Original regressors X (as in the main model).
  - Original regressors plus squared terms and interactions.

Illustration: For `jour_lm`, variance seems to decreases with the fitted values, or, equivalently, increases with `log(citeprice)`.

Null distribution: Approximately χ^2_q, where q is the number of auxiliary regressors (excluding constant term).

Testing for heteroskedasticity

In R: `bptest()` from **lmtest** implements Breusch-Pagan test.

- Provides several flavors of the test via specification of a variance formula.
- Provides studentized version (overcomes assumption of normally distributed errors).
- Default: Studentized version for auxiliary regression with original regressors *X*.

Example using defaults

```
R> bptest(jour_lm)

        studentized Breusch-Pagan test

data:  jour_lm
BP = 9.8, df = 1, p-value = 0.002
```

Testing for heteroskedasticity

White test uses original regressors as well as their squares and interactions in auxiliary regression.

Can use `bptest()`:

```
R> bptest(jour_lm, ~ log(citeprice) + I(log(citeprice)^2),
+    data = journals)

        studentized Breusch-Pagan test

data:  jour_lm
BP = 11, df = 2, p-value = 0.004
```

Testing for heteroskedasticity

Goldfeld-Quandt test:

- Nowadays probably more popular in textbooks than in applied work.
- Order sample with respect to the variable explaining the heteroskedasticity (in example: price per citation).
- Split sample and compare mean residual sum of squares before and after the split point via an F test.
- Problem: A meaningful split point is rarely known in advance.
- Modification: Omit some central observations to improve the power.

Testing for heteroskedasticity

In R: `gqtest()`.

Default: Assume that the data are already ordered, split sample in the middle without omitting any central observations.

Illustration: Order the observations with respect to price per citation.

```
R> gqtest(jour_lm, order.by = ~ citeprice, data = journals)

        Goldfeld-Quandt test

data:  jour_lm
GQ = 1.7, df1 = 88, df2 = 88, p-value = 0.007
alternative hypothesis: variance increases from segment 1 to 2
```

Testing the functional form

Assumption: $\mathsf{E}(\varepsilon|X) = 0$ is crucial for consistency of the least-squares estimator.

Source for violation: Typically, misspecification of the functional form, e.g., by omitting relevant variables.

Ramsey's RESET:

- Regression specification error test.
- Construct auxiliary variables and assess their significance using a simple *F* test.
- Auxiliary variables: Powers of the
  - fitted values $\hat{y}$,
  - original regressors,
  - first principal component of X.

Testing the functional form

In R: `resettest()` provides all three versions.

Default: Second and third powers of the fitted values as auxiliary variables.

Illustration: With only one real regressor in the model matrix X (excluding the intercept), all three strategies yield equivalent results.

```
R> resettest(jour_lm)

        RESET test

data:  jour_lm
RESET = 1.4, df1 = 2, df2 = 180, p-value = 0.2
```

Testing the functional form

Rainbow test:

- Idea: Even a misspecified model might fit (reasonably) well in the "center" of the sample but might lack fit in the tails.
- Fit model to a subsample (typically, the middle 50%) and compare with full sample fit using an *F* test.
- Determination of "middle": ordering by a regressor or by the Mahalanobis distance of the regressor vector x_i to the mean regressor.

In R: `raintest()` implements both flavours (and some further options for the subsample choice).

Default: Assume that the data are already ordered and use middle 50% as subsample.

Testing the functional form

Illustration: Assess stability of functional form over age of journals.

```
R> raintest(jour_lm, order.by = ~ age, data = journals)

	Rainbow test

data:  jour_lm
Rain = 1.8, df1 = 90, df2 = 88, p-value = 0.004
```

Interpretation:

- The fit for the 50% “middle-aged” journals is significantly different from the fit comprising all journals.
- Relationship between the number of subscriptions and the price per citation also depends on the age of the journal.
- As we will see below: Libraries are willing to pay more for established journals.

Testing the functional form

Harvey-Collier test:

- Order sample prior to testing.
- Compute recursive residuals of the fitted model.
- Recursive residuals are essentially standardized one-step-ahead prediction errors.
- If model is correctly specified, recursive residuals have mean zero.
- If mean differs from zero, ordering variable has an influence on the regression relationship.
- Use simple *t* test for testing.

Testing the functional form

In R: `harvtest()`.

Default: Assume that the data are already ordered.

```
R> harvtest(jour_lm, order.by = ~ age, data = journals)

        Harvey-Collier test

data:  jour_lm
HC = 5.1, df = 180, p-value = 9e-07
```

Interpretation: Confirms that age of journals has significant influence on regression relationship.

Testing for autocorrelation

Problem: Time series regressions are often affected by autocorrelation (or serial correlation), just as disturbances in cross-section models are typically heteroskedastic.

Illustration: Reconsider first model for US consumption function.

```
R> library("dynlm")
R> data("USMacroG", package = "AER")
R> consump1 <- dynlm(consumption ~ dpi + L(dpi),
+    data = USMacroG)
```

Testing for autocorrelation

Durbin-Watson test:

- Classical test for autocorrelation in regressions.
- Test statistic: Ratio of the sum of squared first differences of residuals (i.e., $(\hat{\varepsilon}_i - \hat{\varepsilon}_{i-1})^2$), and the RSS.
- Under the null hypothesis of no autocorrelation, test statistic ≈ 2.
- Under the alternative of positive autocorrelation, it typically is much smaller.
- Null distribution is nonstandard: for Gaussian errors, the distribution of a linear combination of χ^2 variables with weights depending on regressor matrix X.
- classical solution: many textbooks still recommend using tabulated upper and lower bounds of critical values.

Testing for autocorrelation

In R: `dwtest()` implements an exact procedure for computing the *p* value (for Gaussian data) and also provides a normal approximation for sufficiently large samples (both depending on the regressor matrix *X*).

```
R> dwtest(consump1)

        Durbin-Watson test

data:  consump1
DW = 0.087, p-value <2e-16
alternative hypothesis: true autocorrelation is greater than 0
```

Interpretation: Highly significant positive autocorrelation, which confirms the results from Chapter 3.

Testing for autocorrelation

Box-Pierce test/Ljung-Box test:

- Originally suggested for diagnostic checking of ARIMA models.
- (Approximate) χ^2 statistics based on estimates of the autocorrelations up to order p.
- Box-Pierce statistic: n times the sum of squared autocorrelations.
- Ljung-Box refinement: squared autocorrelation at lag j is weighted by $(n+2)/(n-j)$.

In R: `Box.test()` in base R (package **stats**) implements both versions.

Default: Box-Pierce test with $p = 1$.

Testing for autocorrelation

Remark: Unlike diagnostic tests in **lmtest**, function expects a series of residuals and not the specification of a linear model as its first argument.

```
R> Box.test(residuals(consump1), type = "Ljung-Box")

        Box-Ljung test

data:  residuals(consump1)
X-squared = 180, df = 1, p-value <2e-16
```

Ljung-Box test confirms significant residual autocorrelation.

Testing for autocorrelation

Breusch-Godfrey test:

- LM test against both AR(p) and MA(p) alternatives.
- Fits auxiliary regression that explains the residuals $\hat{\varepsilon}$ by the original regressors X augmented by the lagged residuals up to order p ($\hat{\varepsilon}_{i-1}, \dots, \hat{\varepsilon}_{i-p}$) (where zeros are used as starting values).
- The resulting RSS is compared with the RSS of the original RSS in a χ^2 (or F) test.
- Works also in the presence of lagged dependent variables (unlike the Durbin-Watson test).

Testing for autocorrelation

In R: `bgtest()` implements both versions.

Default: Use order $p = 1$.

```
R> bgtest(consump1)

        Breusch-Godfrey test for serial correlation of order up
        to 1

data:  consump1
LM test = 190, df = 1, p-value <2e-16
```

Diagnostics and Alternative Methods of Regression

Robust Standard Errors and Tests

Robust standard errors and tests

Starting point: Economic data typically exhibit some form of autocorrelation and/or heteroskedasticity.

Problem: Known covariance structure could be taken into account in a (parametric) model. More often than not, form of the autocorrelation or heteroskedasticity is unknown.

Estimation: OLS typically still consistent.

Inference: For valid inference a consistent covariance matrix estimate is essential.

Solution: Heteroskedasticity consistent (HC) and heteroskedasticity and autocorrelation consistent (HAC) covariance matrix estimators.

Robust standard errors and tests

More specifically: Standard *t* and *F* tests (performed when calling `summary()` or `anova()`) assume that errors are homoskedastic and uncorrelated given the regressors: $\text{Var}(\varepsilon|X) = \sigma^2 I$.

In practice: often $\text{Var}(\varepsilon|X) = \Omega$, where Ω is unknown. Implies

$$\text{Var}(\hat{\beta}|X) \quad = \quad \left(X^\top X\right)^{-1} X^\top \Omega X \left(X^\top X\right)^{-1},$$

Only reduces to $\sigma^2(X^\top X)^{-1}$ if errors are indeed homoskedastic and uncorrelated.

Robust standard errors and tests

In R:

- Package **sandwich** (automatically loaded with **AER**) provides HC and HAC counterparts of `vcov()`: `vcovHC()` and `vcovHAC()`
- Quasi-*t* and quasi-*F* tests based on robust covariances via functions from **lmtest**:
  - `coeftest()`: generalization of `summary()`.
  - `waldtest()`: generalization of `anova()`.

HC estimators

In cross-section regressions: Assume that Ω is a diagonal matrix. A plug-in estimator for Var$(\hat{\beta}|X)$ could use $\hat{\Omega} = \text{diag}(\omega_1, \dots, \omega_n)$ with:

$$
\begin{aligned}
\text{const:} \quad \omega_i &= \hat{\sigma}^2 \\
\text{HC0:} \quad \omega_i &= \hat{\varepsilon}_i^2 \\
\text{HC1:} \quad \omega_i &= \frac{n}{n-k}\,\hat{\varepsilon}_i^2 \\
\text{HC2:} \quad \omega_i &= \frac{\hat{\varepsilon}_i^2}{1-h_{ii}} \\
\text{HC3:} \quad \omega_i &= \frac{\hat{\varepsilon}_i^2}{(1-h_{ii})^2} \\
\text{HC4:} \quad \omega_i &= \frac{\hat{\varepsilon}_i^2}{(1-h_{ii})^{\delta_i}}
\end{aligned}
$$

where h_{ii} are the hat values, $\bar{h}$ is their mean, and $\delta_i = \min\{4, h_{ii}/\bar{h}\}$.

HC estimators

Details:

- Const: Standard estimator for homoskedastic errors.
- HC0: Basic sandwich estimator (Eicker/Huber/White).
- HC1–HC3: Small sample improvements.
- HC4: Improve small-sample performance, especially in the presence of influential observations.

In R: `vcovHC()` computes all versions of covariance estimators from a fitted linear model, just as `vcov()`.

Default: HC3.

HC estimators

Illustration: For journals regression

```
R> vcov(jour_lm)
```

```
                (Intercept) log(citeprice)
(Intercept)       3.126e-03     -6.144e-05
log(citeprice)   -6.144e-05      1.268e-03
```

```
R> vcovHC(jour_lm)
```

```
                (Intercept) log(citeprice)
(Intercept)        0.003085       0.000693
log(citeprice)     0.000693       0.001188
```

HC estimators

Coefficient summary: Regression output typically contains table with regression coefficients, their standard errors, and associated *t* statistics and *p* values.

- `summary(jour_lm)` computes table along with additional information about the model.
- `coeftest(jour_lm)`computes only the table.
- It additionally allows for specification of a `vcov` argument, either as a function or directly as a fitted matrix.

Illustration: Apply all approaches to journals regression (all leading to almost identical results).

HC estimators

```
R> summary(jour_lm)

Call:
lm(formula = log(subs) ~ log(citeprice), data = journals)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.7248 -0.5361  0.0372  0.4662  1.8481 

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)      4.7662     0.0559    85.2   <2e-16
log(citeprice)  -0.5331     0.0356   -15.0   <2e-16

Residual standard error: 0.75 on 178 degrees of freedom
Multiple R-squared:  0.557,	Adjusted R-squared:  0.555 
F-statistic:  224 on 1 and 178 DF,  p-value: <2e-16
```

HC estimators

```
R> coeftest(jour_lm)

t test of coefficients:

               Estimate Std. Error t value Pr(>|t|)
(Intercept)      4.7662     0.0559    85.2   <2e-16
log(citeprice)  -0.5331     0.0356   -15.0   <2e-16
```

HC estimators

```
R> coeftest(jour_lm, vcov = vcovHC)

t test of coefficients:

               Estimate Std. Error t value Pr(>|t|)
(Intercept)      4.7662     0.0555    85.8   <2e-16
log(citeprice)  -0.5331     0.0345   -15.5   <2e-16
```

HC estimators

```
R> coeftest(jour_lm, vcov = vcovHC(jour_lm))

t test of coefficients:

               Estimate Std. Error t value Pr(>|t|)
(Intercept)      4.7662     0.0555    85.8   <2e-16
log(citeprice)  -0.5331     0.0345   -15.5   <2e-16
```

HC estimators

Comparison of the different types of HC estimators:

```
R> t(sapply(c("const", "HC0", "HC1", "HC2", "HC3", "HC4"),
+    function(x) sqrt(diag(vcovHC(jour_lm, type = x)))))
```

```
      (Intercept) log(citeprice)
const     0.05591        0.03561
HC0       0.05495        0.03377
HC1       0.05526        0.03396
HC2       0.05525        0.03412
HC3       0.05555        0.03447
HC4       0.05536        0.03459
```

All estimators lead to almost identical results. In fact, all standard errors are slightly smaller than those computed under the assumption of homoskedastic errors.

HC estimators

Illustration: Reconsider public schools data to show that using robust covariances can make a big difference.

Model: Include quadratic term in model. This appears to be significant due to influential observations (most notably Alaska), but is in fact spurious.

```
R> ps_lm <- lm(Expenditure ~ Income, data = ps)
R> ps_lm2 <- lm(Expenditure ~ Income + I(Income^2), data = ps)
R> anova(ps_lm, ps_lm2)
```

```
Analysis of Variance Table

Model 1: Expenditure ~ Income
Model 2: Expenditure ~ Income + I(Income^2)
  Res.Df    RSS Df Sum of Sq    F Pr(>F)
1     48 181015
2     47 150986  1     30030 9.35 0.0037
```

HC estimators

`waldtest()` provides the same type of test as `anova()` but has a `vcov` argument. `vcov` can be either a function or a matrix (computed for the more complex model).

```
R> waldtest(ps_lm, ps_lm2, vcov = vcovHC(ps_lm2, type = "HC4"))

Wald test

Model 1: Expenditure ~ Income
Model 2: Expenditure ~ Income + I(Income^2)
  Res.Df Df    F Pr(>F)
1     48
2     47  1 0.08   0.77
```

Interpretation: Residuals are not really heteroskedastic, but have influential observations. However, HC estimators also yield appropriate results in this situation, showing that the quadratic term is not significant.

HAC estimators

In time series regressions: If the error terms ε_i are correlated, Ω is not diagonal and can only be estimated directly upon introducing further assumptions on its structure.

Solution: If form of heteroskedasticity and autocorrelation is unknown, valid standard errors and tests may be obtained by estimating $X^\top \Omega X$ instead.

Technically: Computing weighted sums of the empirical autocorrelations of $\hat{\varepsilon}_i x_i$.

Estimators: Differ with respect to choice of weights.

Choice of weights: Based on different kernel functions and bandwidth selection strategies.

HAC estimators

In R: `vcovHAC()` provides general framework for HAC estimators.

Convenience interfaces:

- `NeweyWest()` (by default) uses a Bartlett kernel with nonparametric bandwidth selection (Newey and West 1987, 1994).
- `kernHAC()` (by default) uses a quadratic spectral kernel with parametric bandwidth selection (Andrews 1991, Andrews and Monahan 1992).
- Both use prewhitening (by default).
- `weave()` implements the class of weighted empirical adaptive variance estimators (Lumley and Heagerty 1999).

Illustration: Newey-West and Andrews kernel HAC estimators for consumption function regression.

HAC estimators

Comparison of standard errors: Spherical errors, quadratic spectral kernel and Bartlett kernel HAC estimators (both using prewhitening).

```
R> rbind(SE = sqrt(diag(vcov(consump1))),
+    QS = sqrt(diag(kernHAC(consump1))),
+    NW = sqrt(diag(NeweyWest(consump1))))
```

```
   (Intercept)    dpi L(dpi)
SE       14.51 0.2063 0.2075
QS       94.11 0.3893 0.3669
NW      100.83 0.4230 0.3989
```

Interpretation: Both sets of robust standard errors are rather similar (except maybe for the intercept) and much larger than the uncorrected standard errors.

These can again be passed to `coeftest()` or `waldtest()` (and other inference functions).

Diagnostics and Alternative Methods of Regression

Resistant Regression

Resistant regression

Goal: Regression that is “resistant” (or “robust”) to a (small) group of outlying observations.

Previously: Leave-one-out (deletion) diagnostics.

Problem: Outliers of the same type can mask each other in leave-one-out diagnostics.

Solution: In low-dimensional problems, use plotting. In high-dimensional data, use regressions that withstand alterations in a certain percentage of the data.

Estimators: Least median of squares (LMS) and least trimmed squares (LTS) regression.

Resistant regression

Optimization problems:

$$\text{LMS:} \qquad \operatorname*{arg\,min}_{\beta} \operatorname{med}_i \, \varepsilon_i^2$$

$$\text{LTS:} \qquad \operatorname*{arg\,min}_{\beta} \sum_{i=1}^{q} \hat{\varepsilon}_{i:n}^2(\beta)$$

where $\hat{\varepsilon}_i = y_i - x_i^\top \beta$ and $i : n$ denotes that the residuals are arranged in increasing order.
LTS preferable on theoretical grounds.

In econometrics: not widely used (unfortunately).

Resistant regression

Illustration: Classical textbook Solow model.

$$\log(Y_t/Y_0) = \beta_1 + \beta_2 \log(Y_0) + \beta_3 \log(K_t) + \beta_4 \log(n_t + 0.05) + \varepsilon_t$$

- Y_t and Y_0 are real GDP in periods t and 0.
- K_t is a measure of the accumulation of physical capital.
- n_t is population growth (plus 5% for labor productivity growth).

Data: `OECDGrowth` provides these variables for the period 1960–1985 for all OECD countries with a population exceeding 1 million.

- `gdp85` and `gdp60` are real GDP (per person of working age).
- `invest` is the average of the annual ratios of real domestic investment to real GDP.
- `popgrowth` is annual population growth.

Resistant regression

OLS estimation:

- Reasonable fit for a cross-section regression.
- Coefficients on `gdp60` and `invest` highly significant.
- Coefficient on `popgrowth` is borderline at 10% level.

In R:

```
R> data("OECDGrowth", package = "AER")
R> solow_lm <- lm(log(gdp85/gdp60) ~ log(gdp60) +
+    log(invest) + log(popgrowth + .05), data = OECDGrowth)
```

Resistant regression

```
R> summary(solow_lm)

Call:
lm(formula = log(gdp85/gdp60) ~ log(gdp60) + log(invest) +
  log(popgrowth + 0.05), data = OECDGrowth)

Residuals:
    Min      1Q  Median      3Q     Max 
-0.1840 -0.0399 -0.0078  0.0451  0.3188 

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)             2.9759     1.0216    2.91   0.0093
log(gdp60)             -0.3429     0.0565   -6.07  9.8e-06
log(invest)             0.6501     0.2020    3.22   0.0048
log(popgrowth + 0.05)  -0.5730     0.2904   -1.97   0.0640

Residual standard error: 0.133 on 18 degrees of freedom
Multiple R-squared:  0.746,	Adjusted R-squared:  0.704 
F-statistic: 17.7 on 3 and 18 DF,  p-value: 1.34e-05
```

Resistant regression

Detection of outliers:

- Graphical displays are not as effective with three regressors.
- Instead: First run LTS analysis.
- Flag observations with unusually large residuals.
- Run standard OLS regression excluding the outlying observations.
- LTS may flag too many points as outlying.
- Exclude only bad leverage points $=$ high-leverage points with large LTS residuals.

In R: `lqs()` from package **MASS** (accompanying Venables and Ripley 2002) implements least quantile of squares including LMS, LTS, and other versions.

Default: LTS with $q = \lfloor n/2 \rfloor + \lfloor (k+1)/2 \rfloor$. (Here: $q = 13$.)

Resistant regression

```
R> library("MASS")
R> solow_lts <- lqs(log(gdp85/gdp60) ~ log(gdp60) +
+    log(invest) + log(popgrowth + .05), data = OECDGrowth,
+    psamp = 13, nsamp = "exact")
```

Algorithmic details:

- Setting `psamp = 13` and `nsamp = "exact"` specifies that all conceivable subsamples of size 13 are used.
- Assures that LTS optimization is exactly solved (for $q = 13$).
- Only feasible for small samples.
- Otherwise some other sampling technique should be used (available in `lqs()`).

Resistant regression

Scale estimates: `lqs()` provides two estimates

- The first is defined via the fit criterion.
- The second is based on the variance of those residuals whose absolute value is less than 2.5 times the initial estimate.
- Second estimate is typically used for scaled residuals.

Outliers: Observations with “large” scaled residuals (exceeding 2.5 in absolute values).

```
R> smallresid <- which(
+    abs(residuals(solow_lts)/solow_lts$scale[2]) <= 2.5)
```

High leverage: For consistency, use robust measure of leverage based on robust covariance estimator, e.g., minimum-volume ellipsoid (MVE) or minimum covariance determinant (MCD) estimator.

Resistant regression

In R: `cov.rob()` from **MASS** provides both (default: MVE).

```
R> X <- model.matrix(solow_lm)[,-1]
R> Xcv <- cov.rob(X, nsamp = "exact")
R> nohighlev <- which(
+    sqrt(mahalanobis(X, Xcv$center, Xcv$cov)) <= 2.5)
```

Details:

- Extract model matrix.
- Estimate its covariance matrix by MVE.
- Compute the leverage utilizing the `mahalanobis()` function.
- Store observations that are not high-leverage points.

Resistant regression

Good observations: Observations that have at least one of the desired properties, small residual or low leverage.

```
R> goodobs <- unique(c(smallresid, nohighlev))
```

Bad observations:

```
R> rownames(OECDGrowth)[-goodobs]
```

```
[1] "Canada"    "USA"       "Turkey"    "Australia"
```

Robust OLS: exclude bad leverage points

```
R> solow_rob <- update(solow_lm, subset = goodobs)
```

Resistant regression

```
R> summary(solow_rob)

Call:
lm(formula = log(gdp85/gdp60) ~ log(gdp60) + log(invest) +
  log(popgrowth + 0.05), data = OECDGrowth,
  subset = goodobs)

Residuals:
     Min       1Q   Median       3Q      Max
-0.15454 -0.05548 -0.00651  0.03159  0.26773

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)             3.7764     1.2816    2.95   0.0106
log(gdp60)             -0.4507     0.0569   -7.93  1.5e-06
log(invest)             0.7033     0.1906    3.69   0.0024
log(popgrowth + 0.05)  -0.6504     0.4190   -1.55   0.1429

Residual standard error: 0.107 on 14 degrees of freedom
Multiple R-squared:  0.853,        Adjusted R-squared:  0.822
F-statistic: 27.1 on 3 and 14 DF,  p-value: 4.3e-06
```

Resistant regression

Interpretation:

- Results somewhat different from full-sample OLS.
- Population growth does not seem to belong in this model.
- Population growth does not seem to explain economic growth for this subset of countries (given other regressors).
  Potential explanation: OECD countries fairly homogeneous with respect to population growth, some countries with substantial population growth have been excluded in robust fit.

Extended versions of the Solow model could include further regressors such as human capital (`log(school)`) and technological know-how (`log(randd)`).

Diagnostics and Alternative Methods of Regression

Quantile Regression

Quantile regression

Ideas:

- Least-squares regression can be viewed as a method for modeling the conditional mean of a response.
- Sometimes other characteristics of the conditional distribution more interesting, e.g. median.
- More generally: Model (and compare) quantiles of the response.

See Koenker (2005) for a comprehensive treatment.

The (linear) quantile regression model is given by the conditional quantile functions (indexed by the quantile τ)

$$Q_y(\tau|x) = x_i^\top \beta;$$

$Q_y(\tau|x)$ denotes the τ-quantile of y conditional on x.

Quantile regression

Optimization: Linear programming problem. Estimate β by

$$\arg\min_{\beta} \sum_{i} \varrho_\tau(y_i - x_i^\top \beta)$$

where

$$\varrho_\tau(u) = u\{\tau - I(u < 0)\}, \qquad \tau \in (0,1)$$

I is indicator function.

In R: `rq()` in package **quantreg**.

Illustration: Reconsider `CPS1988` data and quantile versions of a Mincer-type wage equation.

$$\begin{aligned} Q_{\mathtt{log(wage)}}(\tau|x) \quad &= \quad \beta_1 + \beta_2\,\mathtt{experience} + \\ &\quad\quad \beta_3\,\mathtt{experience}^2 + \beta_4\,\mathtt{education} \end{aligned}$$

Quantile regression

Default: `rq()` by default uses $\tau = 0.5$; i.e., median or LAD (for “least absolute deviations”) regression.

LAD regression:

```
R> library("quantreg")
R> data("CPS1988", package = "AER")
R> cps_f <- log(wage) ~ experience + I(experience^2) + education
R> cps_lad <- rq(cps_f, data = CPS1988)
R> summary(cps_lad)
```

```
Call: rq(formula = cps_f, data = CPS1988)

tau: [1] 0.5

Coefficients:
                Value     Std. Error t value   Pr(>|t|)
(Intercept)       4.24088   0.02190  193.67805   0.00000
experience        0.07744   0.00115   67.50041   0.00000
I(experience^2)  -0.00130   0.00003  -49.97891   0.00000
education         0.09429   0.00140   67.57171   0.00000
```

Quantile regression

Particularly useful: Model and compare several quantiles simultaneously.

Illustration: Model first and third quartiles (i.e., $\tau = 0.25$ and $\tau = 0.75$).

```
R> cps_rq <- rq(cps_f, tau = c(0.25, 0.75), data = CPS1988)
```

Question: Are the regression lines or surfaces parallel (i.e., are effects uniform across quantiles)?

Answer: Compare fits with `anova()` method, either in an overall test of all coefficients or in coefficient-wise comparisons.

Quantile regression

```
R> summary(cps_rq)
Call: rq(formula = cps_f, tau = c(0.25, 0.75), data = CPS1988)

tau: [1] 0.25

Coefficients:
                Value    Std. Error t value   Pr(>|t|)
(Intercept)       3.78227   0.02866 131.95189   0.00000
experience        0.09156   0.00152  60.26474   0.00000
I(experience^2)  -0.00164   0.00004 -45.39065   0.00000
education         0.09321   0.00185  50.32520   0.00000

Call: rq(formula = cps_f, tau = c(0.25, 0.75), data = CPS1988)

tau: [1] 0.75

Coefficients:
                Value    Std. Error t value   Pr(>|t|)
(Intercept)       4.66005   0.02023 230.39734   0.00000
experience        0.06377   0.00097  65.41364   0.00000
I(experience^2)  -0.00099   0.00002 -44.15591   0.00000
education         0.09434   0.00134  70.65855   0.00000
```

Quantile regression

```
R> cps_rq25 <- rq(cps_f, tau = 0.25, data = CPS1988)
R> cps_rq75 <- rq(cps_f, tau = 0.75, data = CPS1988)
R> anova(cps_rq25, cps_rq75)

Quantile Regression Analysis of Deviance Table

Model: log(wage) ~ experience + I(experience^2) + education
Joint Test of Equality of Slopes: tau in {  0.25 0.75  }

  Df Resid Df F value Pr(>F)
1  3    56307     115 <2e-16

R> anova(cps_rq25, cps_rq75, joint = FALSE)

Quantile Regression Analysis of Deviance Table

Model: log(wage) ~ experience + I(experience^2) + education
Tests of Equality of Distinct Slopes: tau in {  0.25 0.75  }

                Df Resid Df F value Pr(>F)
experience       1    56309  339.41 <2e-16
I(experience^2)  1    56309  329.74 <2e-16
education        1    56309    0.35   0.55
```

Quantile regression

Visualization: For each regressor: plot estimate as a function of the quantile.

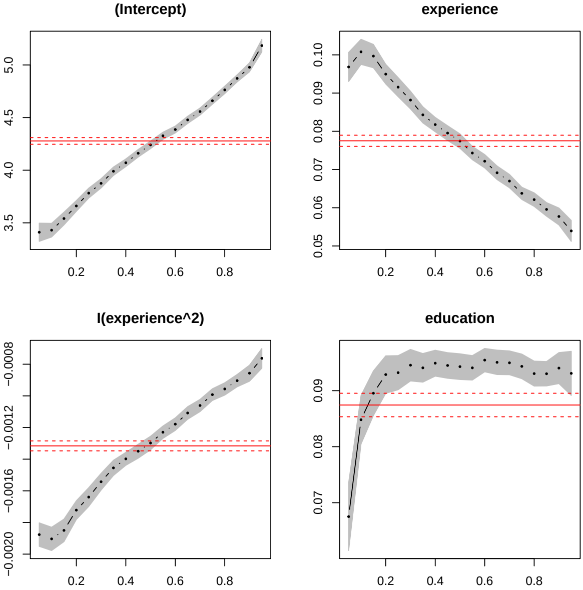
In R: `plot()` method for `summary()` of quantile regression object (for a larger set of quantiles).

```
R> cps_rqbig <- rq(cps_f, tau = seq(0.05, 0.95, by = 0.05),
+    data = CPS1988)
R> cps_rqbigs <- summary(cps_rqbig)
R> plot(cps_rqbigs)
```

Details

- Influence of the covariates clearly not uniform.
- Shaded areas represent pointwise 90% (by default) confidence intervals for the quantile regression estimates.
- For comparison: OLS estimate and associated 90% confidence interval.

Quantile regression



Quantile regression

quantreg also contains

- Nonlinear and nonparametric quantile modeling functions.
- Several algorithms for fitting models (specifically, both exterior and interior point methods).
- Several choices of methods for computing confidence intervals and related test statistics.
- Quantile regression for censored dependent variables (with various fitting algorithms).