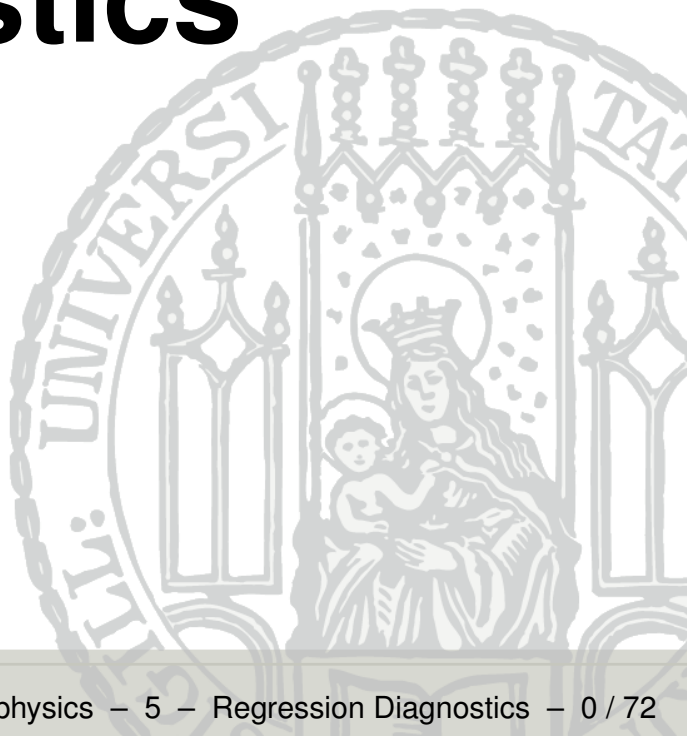
LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Statistical Geophysics

Chapter 5

Regression Diagnostics



Regression Diagnostics

Contents

Contents

- Classical Linear Regression
- Munich Rent Data
- Partial Residuals
- Spatial Covariates
- More Diagnostics
- Model Selection

Regression Diagnostics

Classical Linear Regression

Classical Linear Regression

Recall the linear regression model. For $i = 1, \ldots, n$:

$$\begin{aligned} y_i &= \beta_0 + \beta_1 x_{i1} + \ldots + \beta_k x_{ik} + \varepsilon_i \\ &= \beta_0 + \sum_{j=1}^{k} \beta_j x_{ij} + \varepsilon_i \\ &= \mathbf{x}_i^\top \boldsymbol{\beta} + \varepsilon_i. \end{aligned}$$

In matrix notation:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

and $\boldsymbol{\varepsilon} \sim N(\mathbf{0}, \sigma^2 \mathbf{I})$.

Classical Linear Regression

Least squares: minimize

$$LS(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left(y_i - \mathbf{x}_i^\top \boldsymbol{\beta} \right)^2 = \boldsymbol{\varepsilon}^\top \boldsymbol{\varepsilon}.$$

Matrix notation:

$$\mathbf{X} = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1k} \\ 1 & x_{21} & \cdots & x_{2k} \\ \vdots & \vdots & & \vdots \\ 1 & x_{n1} & \cdots & x_{nk} \end{pmatrix} \quad \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{pmatrix} \quad \boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}.$$

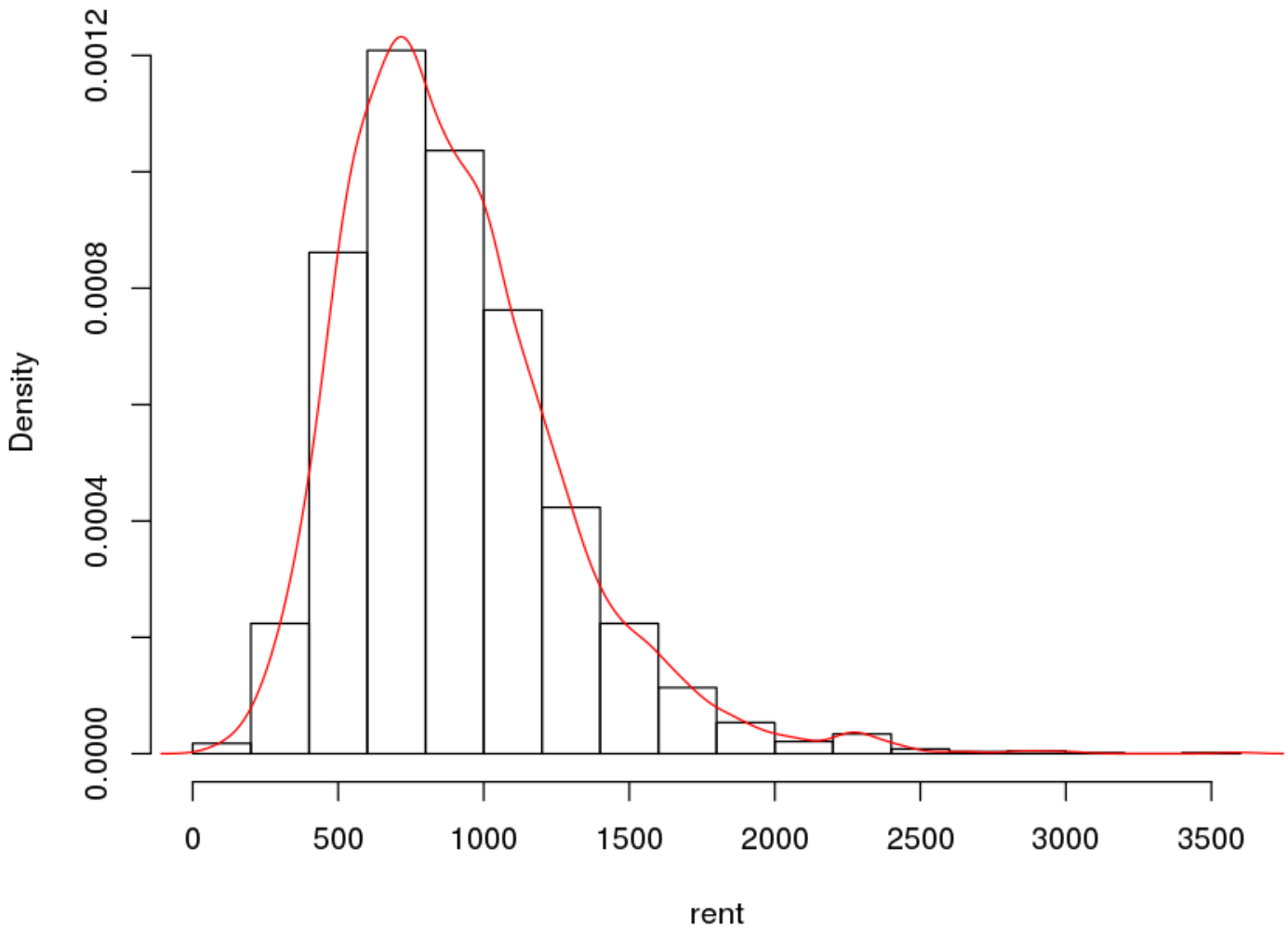
Minimize $||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 \Rightarrow \mathbf{X}^\top \mathbf{X} \hat{\boldsymbol{\beta}} = \mathbf{X}^\top \mathbf{y} \Rightarrow \hat{\boldsymbol{\beta}} = \left(\mathbf{X}^\top \mathbf{X} \right)^{-1} \mathbf{X}^\top \mathbf{y}.$

Munich Rent Data

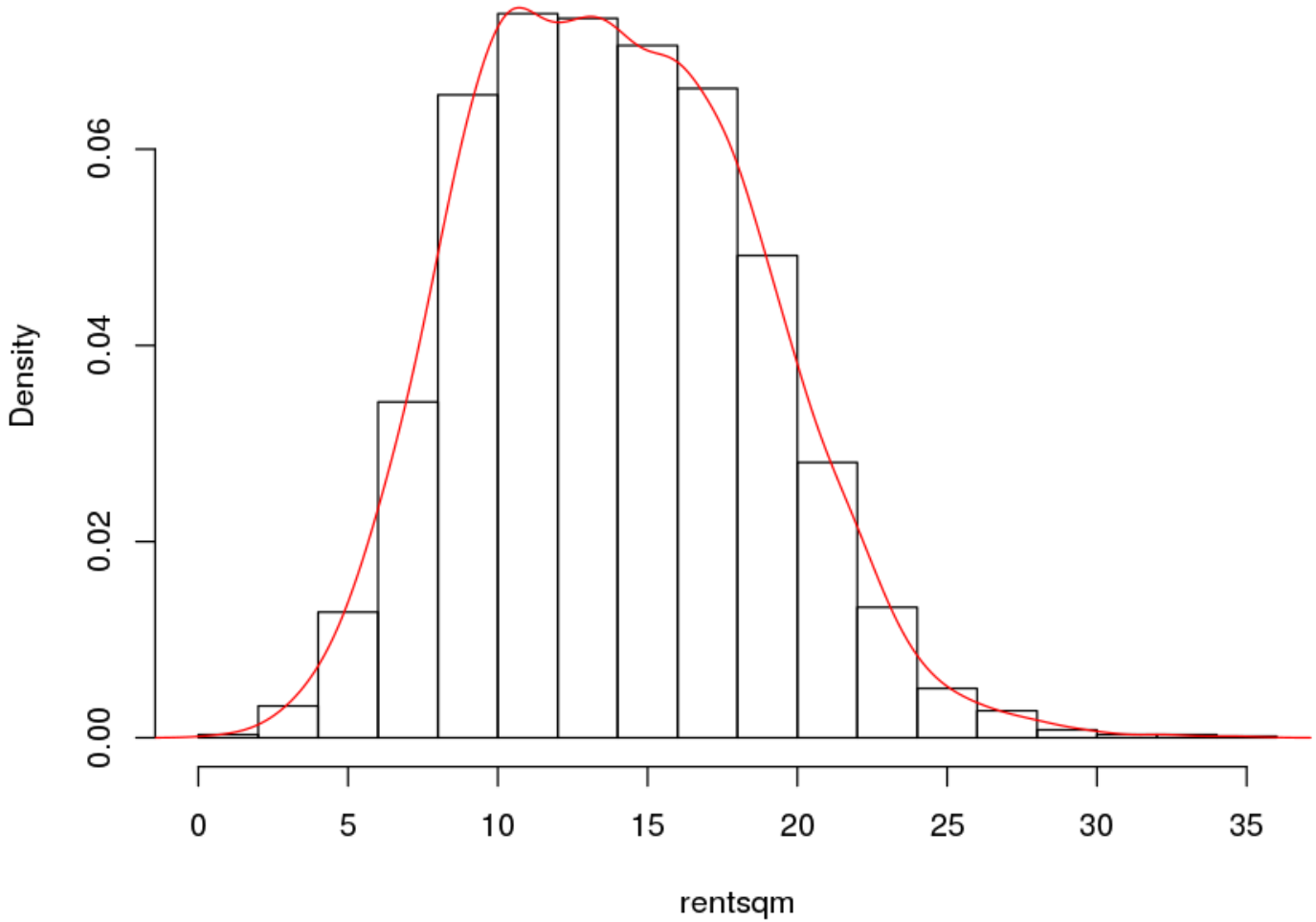
The aim is to establish a rent index to provide information on the “typical rent for a flat”.

Variable	Description.
`rent`	Net rent per month (EUR).
`rentsqm`	Net rent per month per square meter (EUR).
`area`	Living area in square meters.
`yearc`	Year of construction.
`location`	Quality of location: `"average"`, `"good"`, `"top"`.
`bath`	Quality of the bathroom: `"standard"`, `"premium"`.
`kitchen`	Quality of the kitchen: `"standard"`, `"premium"`.
`cheating`	Central heating system: `"yes"`, `"no"`.
`district`	District in Munich.

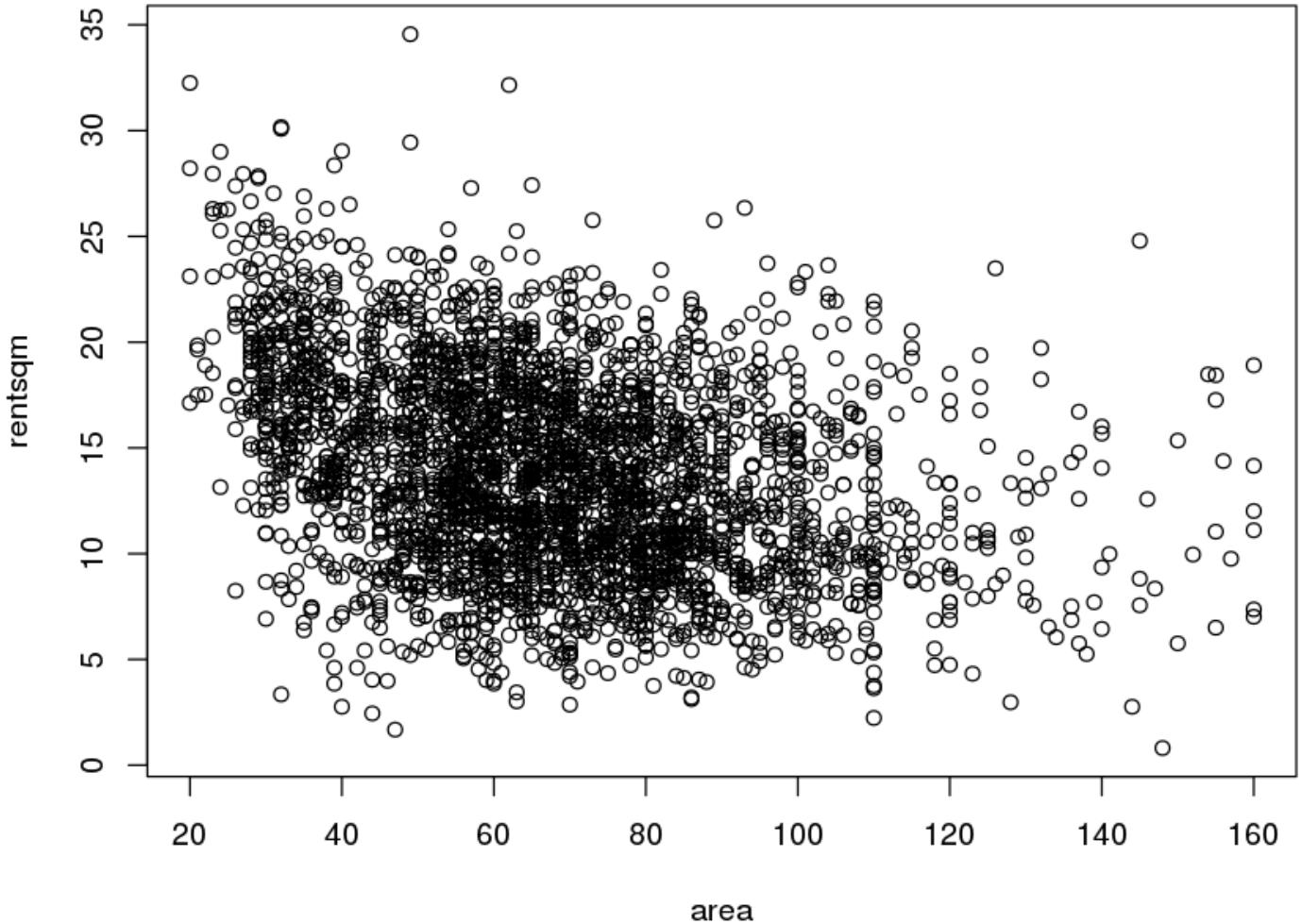
Munich Rent Data



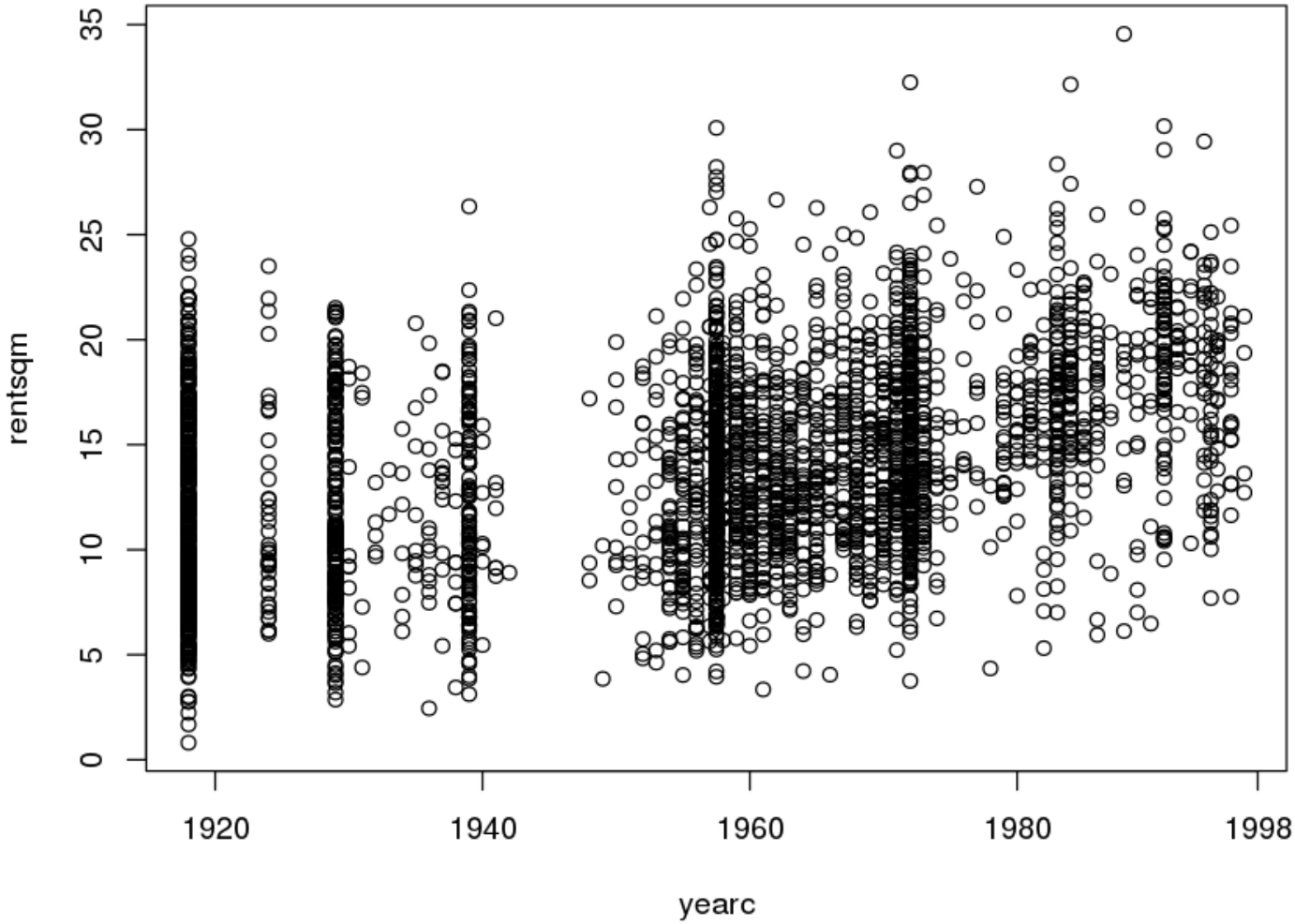
Munich Rent Data



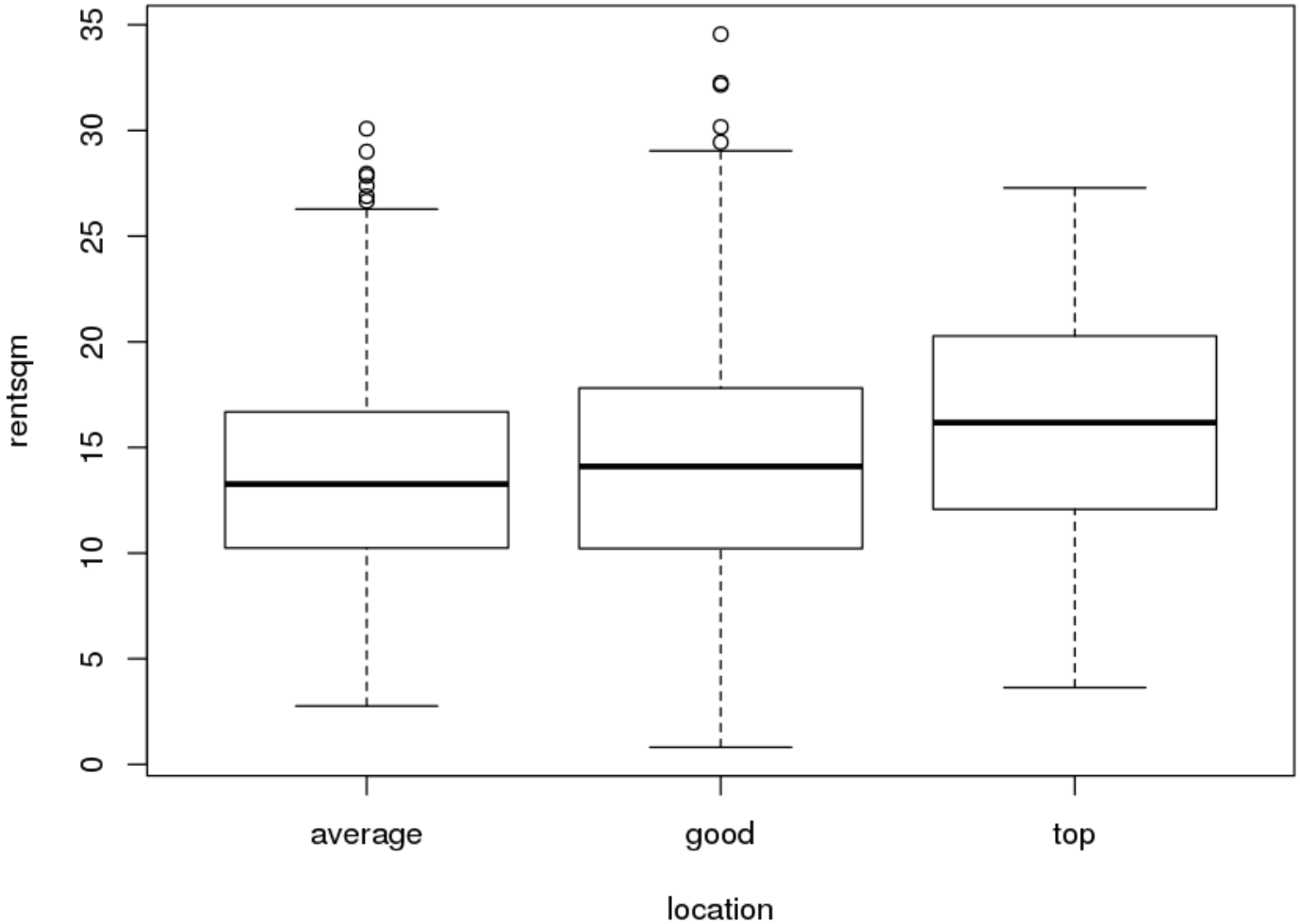
Munich Rent Data



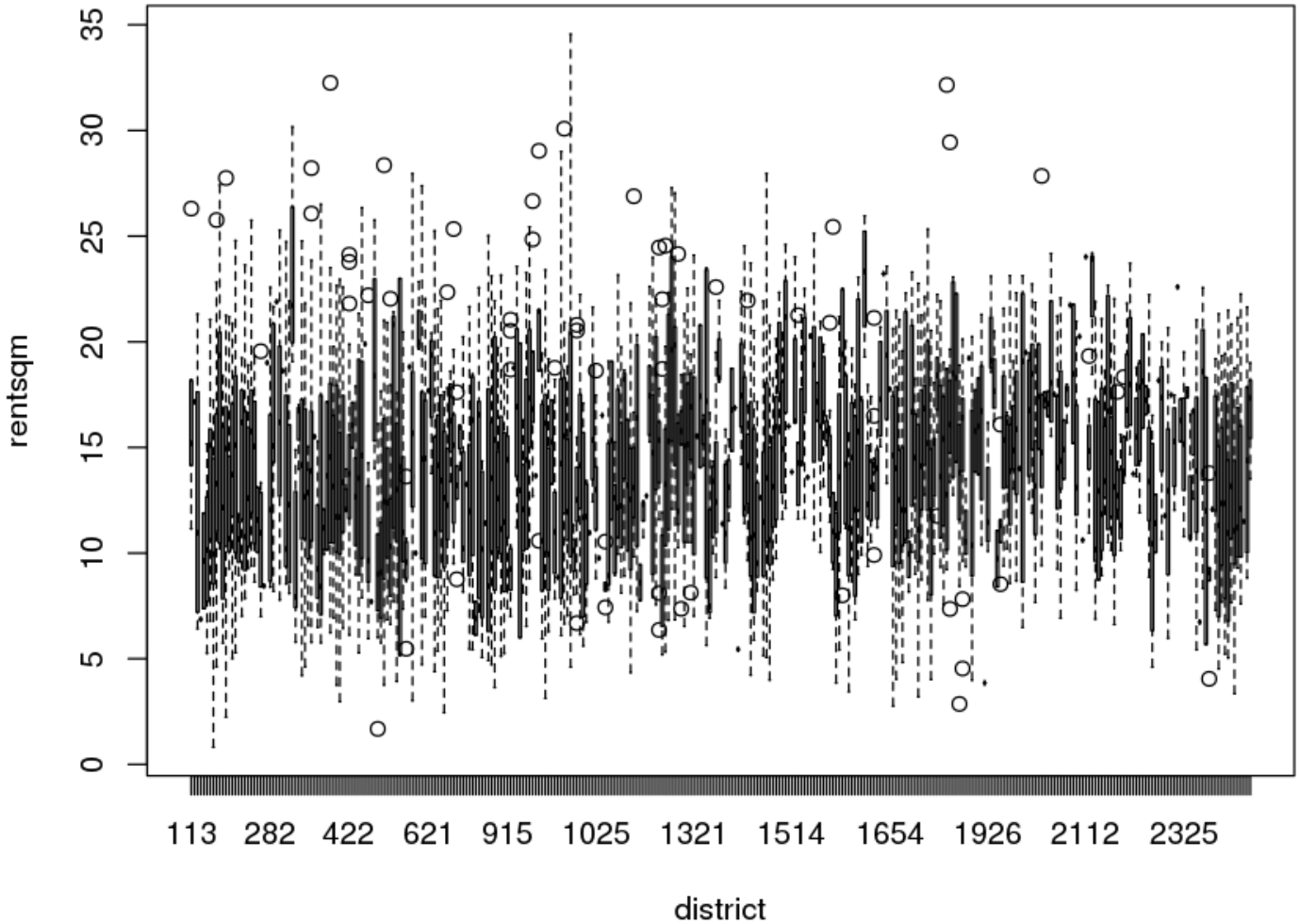
Munich Rent Data



Munich Rent Data



Munich Rent Data



Munich Rent Data

A simple linear regression.

```
R> data(rent99, package = "bamlss")
R> b1 <- lm(rentsqm ~ area + yearc + bath +
+    kitchen + cheating + location, data = rent99)
```

Munich Rent Data

```
R> summary(b1)

Call:
lm(formula = rentsqm ~ area + yearc + bath + kitchen + cheating +
    location, data = rent99)

Residuals:
     Min       1Q   Median       3Q      Max
-12.5390  -2.7556  -0.2092   2.5825  16.8582

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)   -88.677193   7.027361 -12.619  < 2e-16 ***
area           -0.063044   0.003214 -19.618  < 2e-16 ***
yearc           0.052569   0.003599  14.606  < 2e-16 ***
bathpremium     1.487475   0.307240   4.841 1.35e-06 ***
kitchenpremium  2.216971   0.357021   6.210 6.02e-10 ***
cheatingyes     3.442259   0.251683  13.677  < 2e-16 ***
locationgood    1.515409   0.149897  10.110  < 2e-16 ***
locationtop     3.363883   0.460321   7.308 3.45e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.96 on 3074 degrees of freedom
Multiple R-squared:  0.3065, Adjusted R-squared:  0.3049
F-statistic: 194.1 on 7 and 3074 DF,  p-value: < 2.2e-16
```

Partial Residuals

Useful tool for exploring whether the influence of x_j is modeled correctly.

The partial residuals regarding covariate x_j are defined by

$$\hat{\varepsilon}_{x_j,i} = y_i - \hat{\beta}_0 - \hat{\beta}_{j-1} x_{i,j-1} - \hat{\beta}_{j+1} x_{i,j+1} - \ldots - \hat{\beta}_k x_{xik} = \hat{\varepsilon}_i + \hat{\beta}_j x_{ij}.$$

In the partial residuals $\hat{\varepsilon}_{x_j,i}$, all covariate effects with exception of the one associated with x_j are removed.

Partial Residuals

Partial residuals for term `area` in model `b1`.

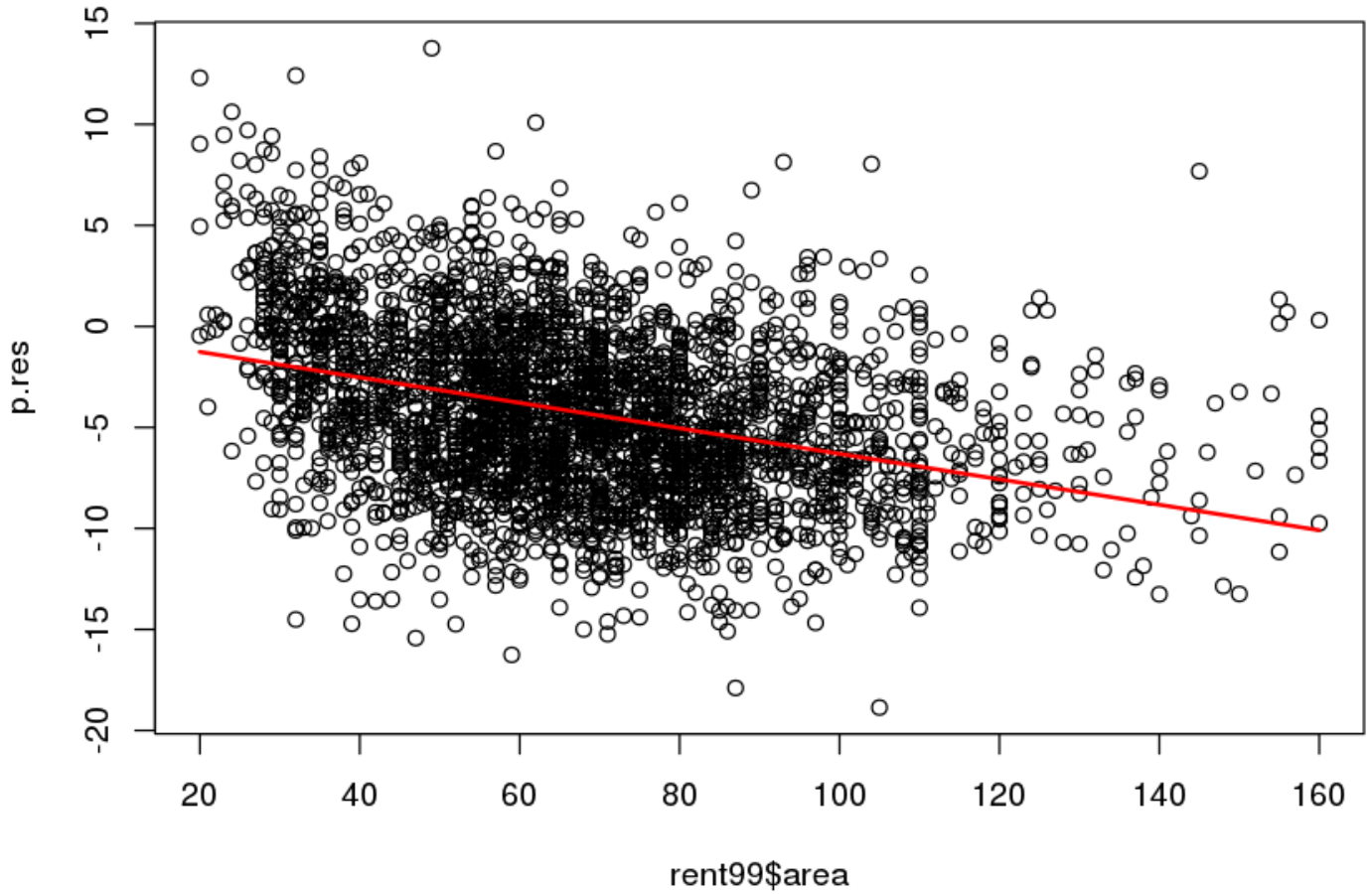
$$\begin{aligned}\hat{\varepsilon}_{\texttt{area},i} &= \texttt{rentsqm}_i - \hat{\beta}_0 - \hat{\beta}_3\texttt{yearc}_i - \ldots - \hat{\beta}_8\texttt{locationtop}_i \\ &= \hat{\varepsilon}_i + \hat{\beta}_1\texttt{area}_i\end{aligned}$$

In R:

```
R> res <- residuals(b1)
R> b2 <- coef(b1)["area"]
R> farea <- b2 * rent99$area
R> p.res <- res + farea
```

Partial Residuals

```
R> plot(p.res ~ rent99$area)
R> i <- order(rent99$area)
R> lines(farea[i] ~ rent99$area[i], col = "red", lwd = 2)
```



Partial Residuals

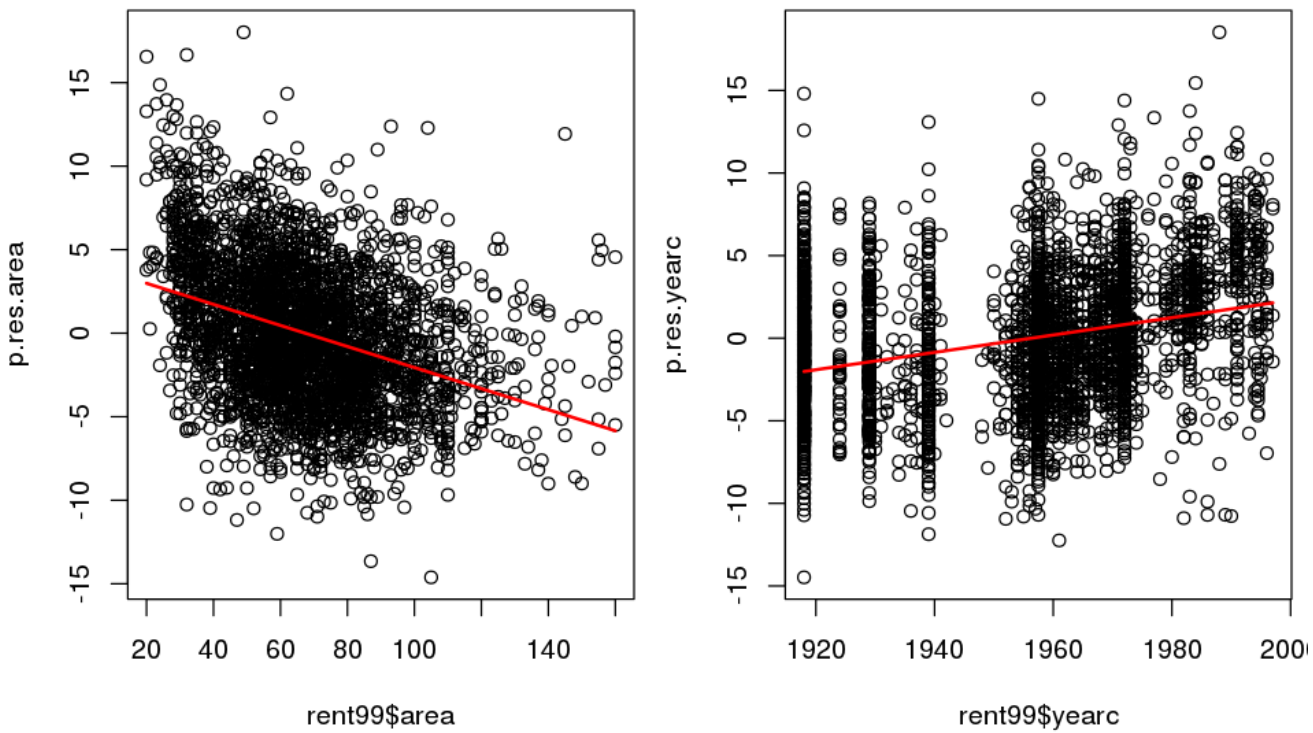
For comparison of multiple effects, it is often useful to require that all functions are "centered around zero". For the `area` effect this can be done, e.g., by

$$\hat{f}(\texttt{area}) = \hat{f}(\texttt{area}) - \overline{\hat{f}(\texttt{area})}.$$

```
R> farea <- b2 * rent99$area
R> farea <- farea - mean(farea)
R> sum(farea)

[1] 9.992007e-13

R> p.res.area <- res + farea

R> fyearc <- coef(b1)["yearc"] * rent99$yearc
R> fyearc <- fyearc - mean(fyearc)
R> sum(fyearc)

[1] -1.105604e-11

R> p.res.yearc <- res + fyearc
```

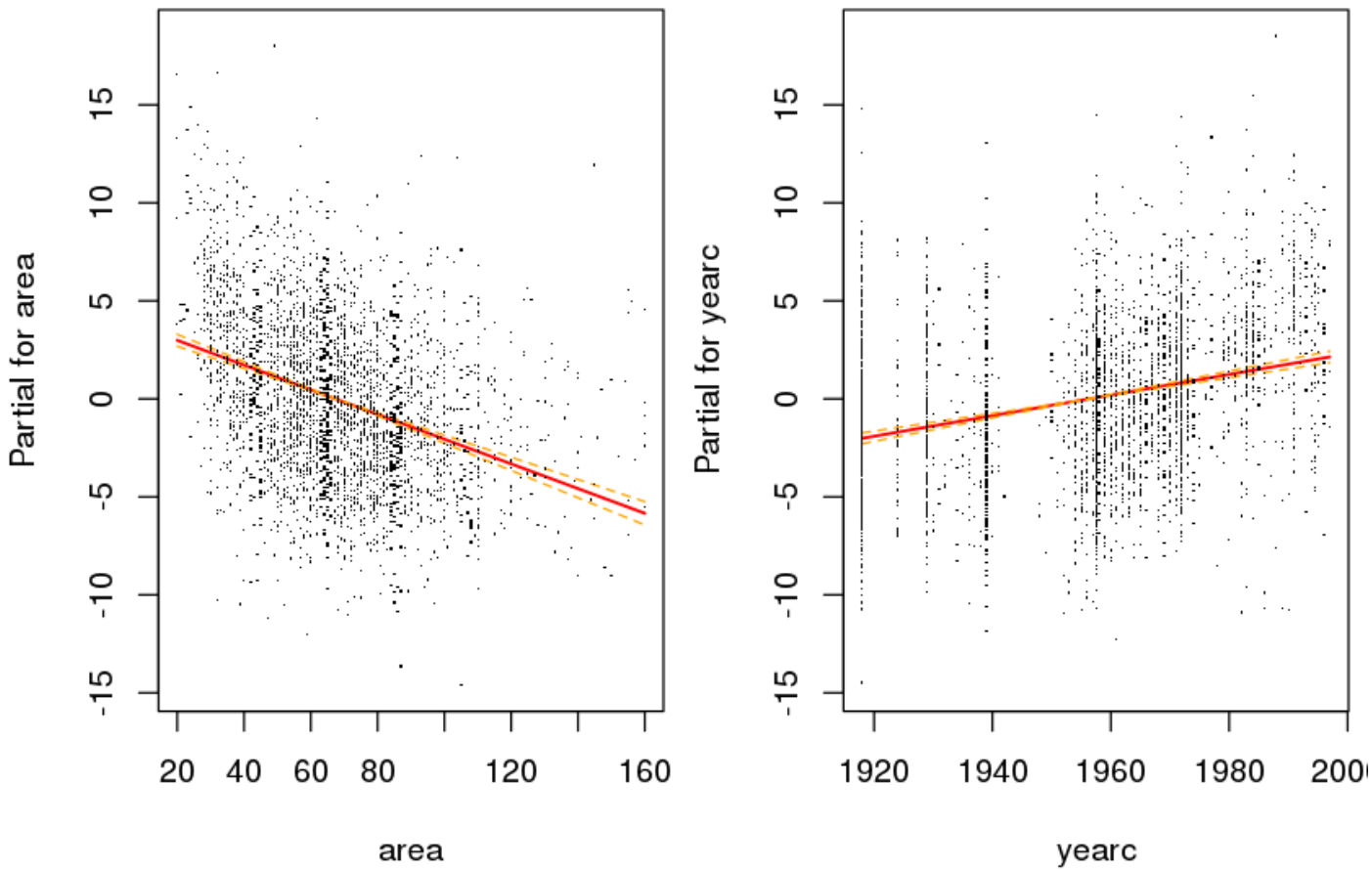
Partial Residuals

```
R> par(mfrow = c(1, 2))
R> j <- order(rent99$yearc)
R> plot(p.res.area ~ rent99$area)
R> lines(farea[i] ~ rent99$area[i], col = "red", lwd = 2)
R> plot(p.res.yearc ~ rent99$yearc)
R> lines(fyearc[j] ~ rent99$yearc[j], col = "red", lwd = 2)
```



Partial Residuals

```
R> par(mfrow = c(1, 2))
R> termplot(b1, term = 1:2, se = TRUE, partial.resid = TRUE,
+    col.res = "black", pch = ".")
```



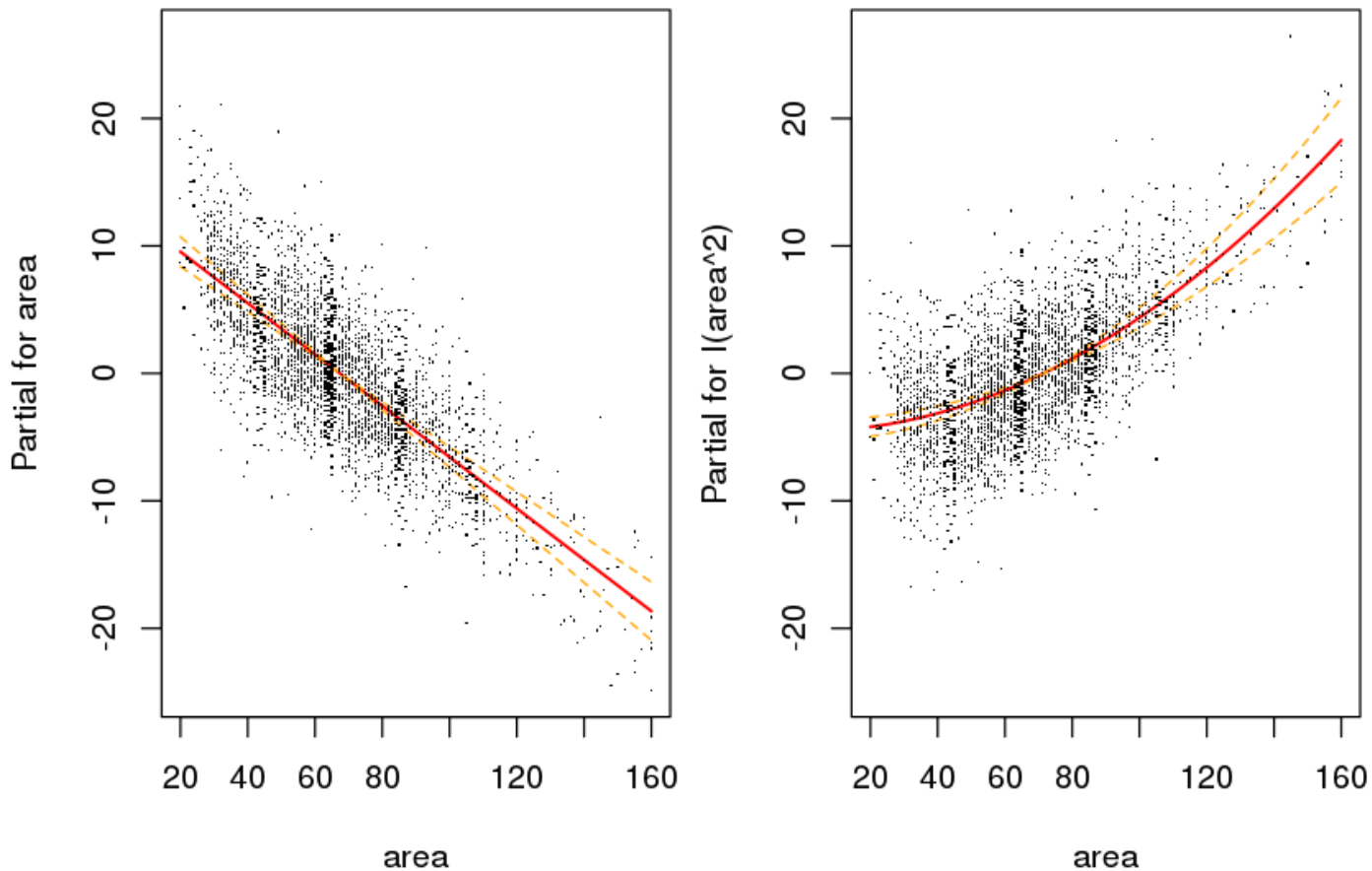
Partial Residuals

Quadratic functions.

```
R> b2 <- lm(rentsqm ~ area + I(area^2) + yearc + I(yearc^2) +
+    bath + kitchen + cheating + location, data = rent99)
```

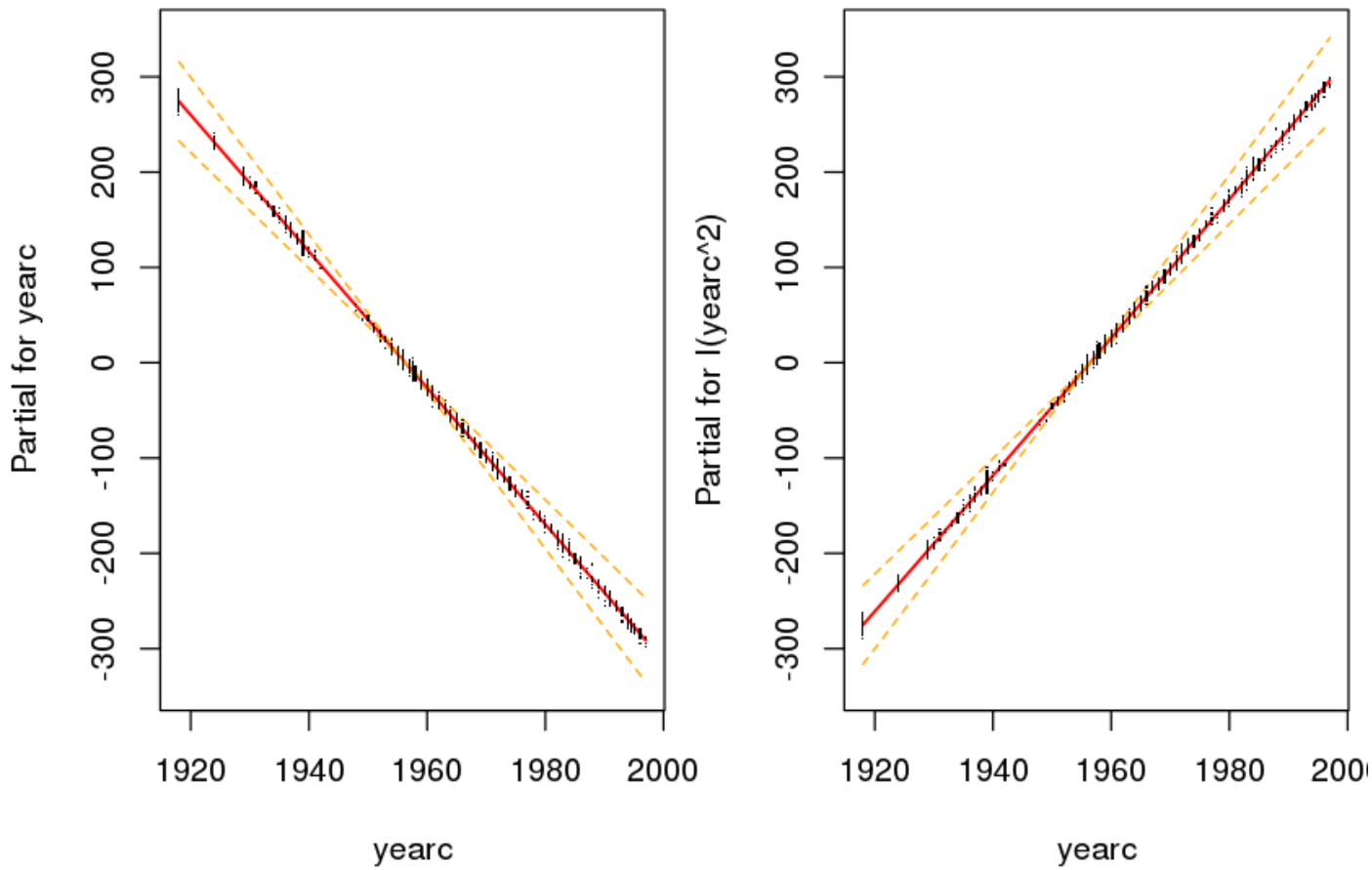
Partial Residuals

```
R> par(mfrow = c(1, 2))
R> termplot(b2, term = 1:2, se = TRUE, partial.resid = TRUE,
+     col.res = "black", pch = ".")
```



Partial Residuals

```
R> par(mfrow = c(1, 2))
R> termplot(b2, term = 3:4, se = TRUE, partial.resid = TRUE,
+     col.res = "black", pch = ".")
```



Partial Residuals

Replace polynomials by so-called orthognal polynomials.

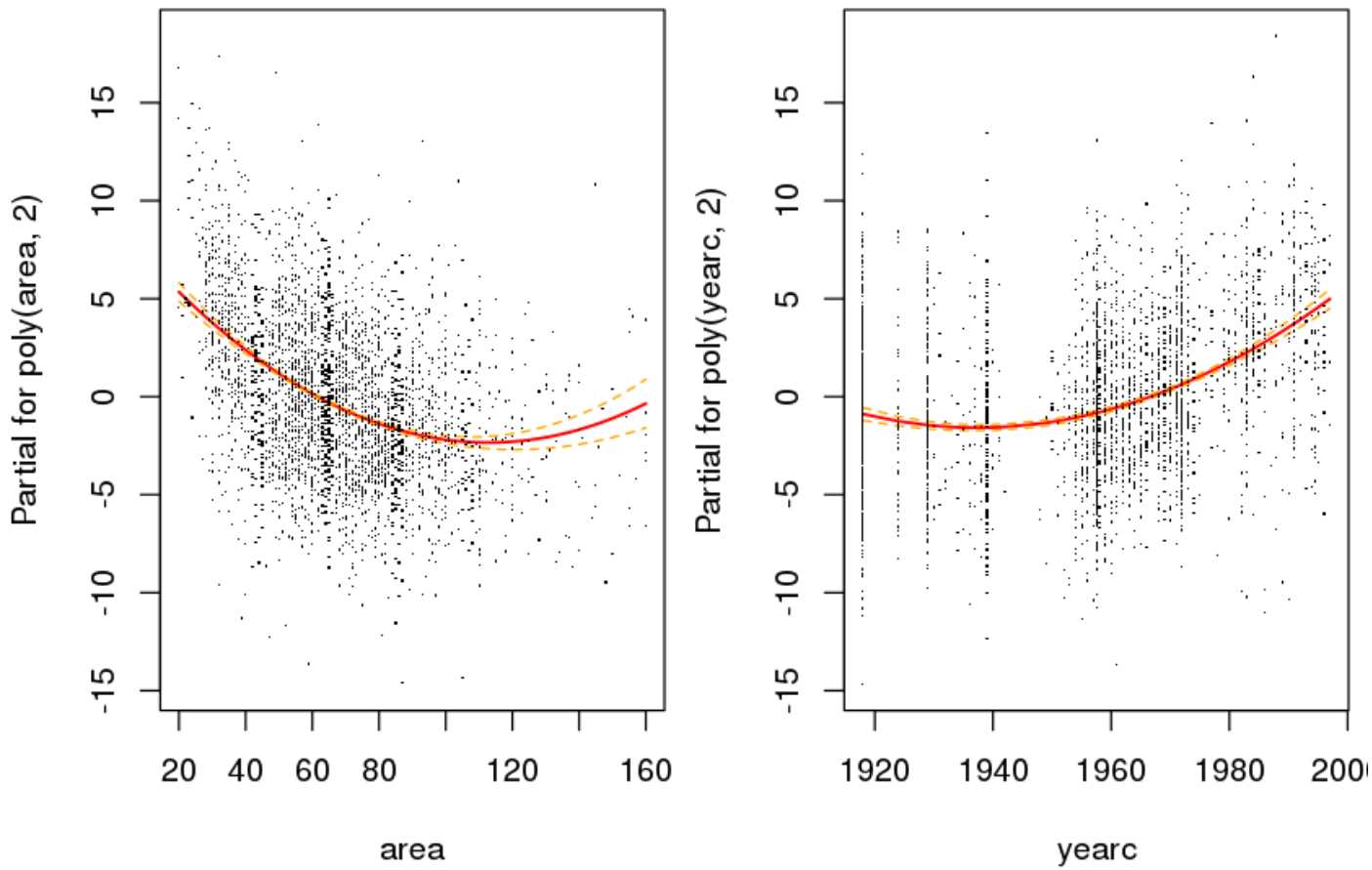
This implies that the columns of the design matrix $\mathbf{X}$ corresponding to, e.g., variable `area` are centered and orthogonal, which leads to stable computation of the least-squares estimator.

In R use function `poly()`:

```
R> b2 <- lm(rentsqm ~ poly(area, 2) + poly(yearc, 2) +
+    bath + kitchen + cheating + location, data = rent99)
```

Partial Residuals

```
R> par(mfrow = c(1, 2))
R> termplot(b2, term = 1:2, se = TRUE, partial.resid = TRUE,
+    col.res = "black", pch = ".")
```



Spatial Covariates

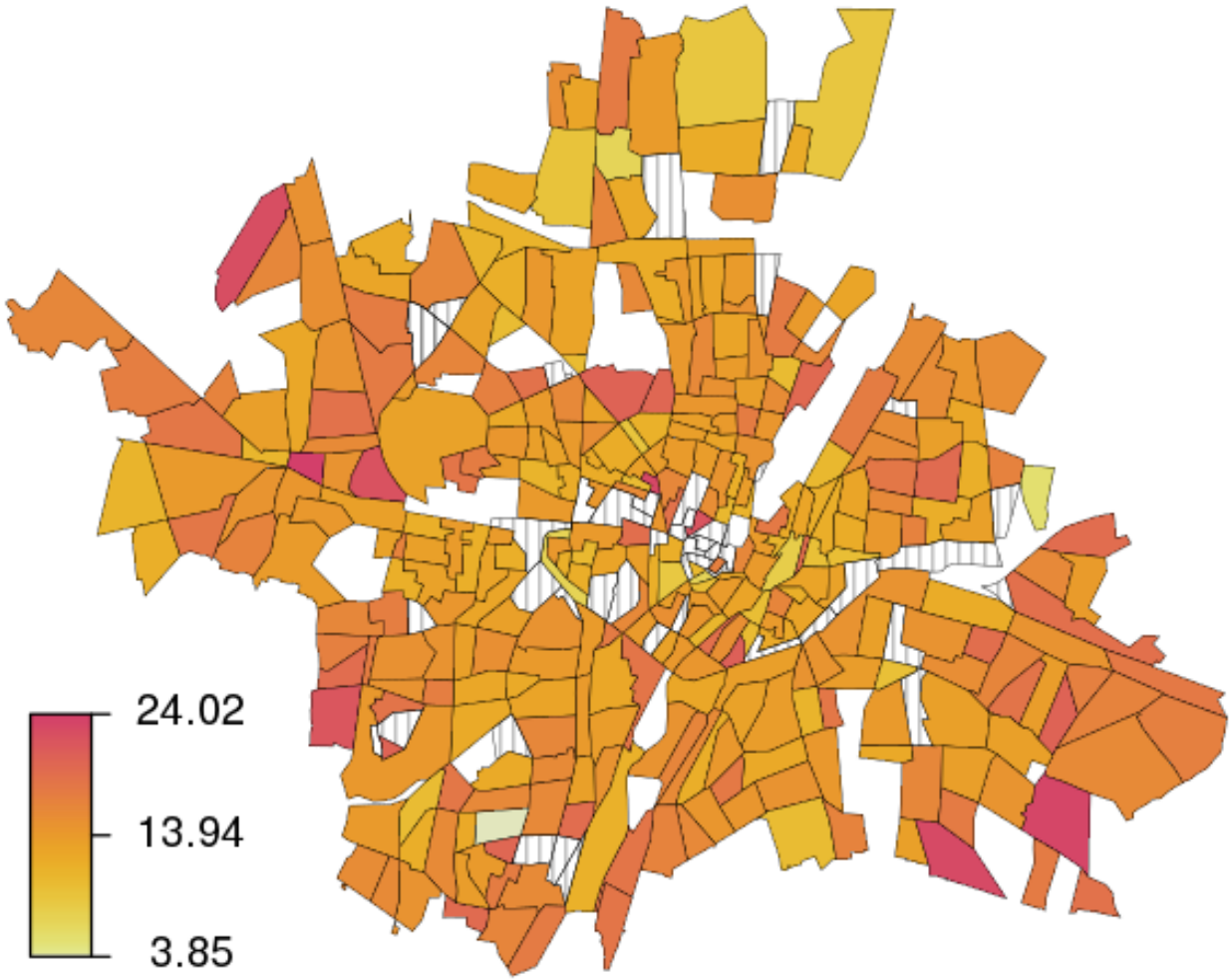
Visualization of spatial variation using function `plotmap()`.

```
R> md <- with(rent99, aggregate(rentsqm,
+    by = list(as.factor(district)),
+    FUN = mean, na.rm = TRUE))
R> head(md)
```

```
  Group.1         x
1     113 16.284670
2     121 17.160000
3     122 12.417570
4     124  6.870833
5     125  9.632732
6     133 10.191534
```

```
R> data("MunichBnd", package = "bamlss")
R> plotmap(MunichBnd, x = md, col = heat_hcl,
+    symmetric = FALSE, swap = TRUE)
```

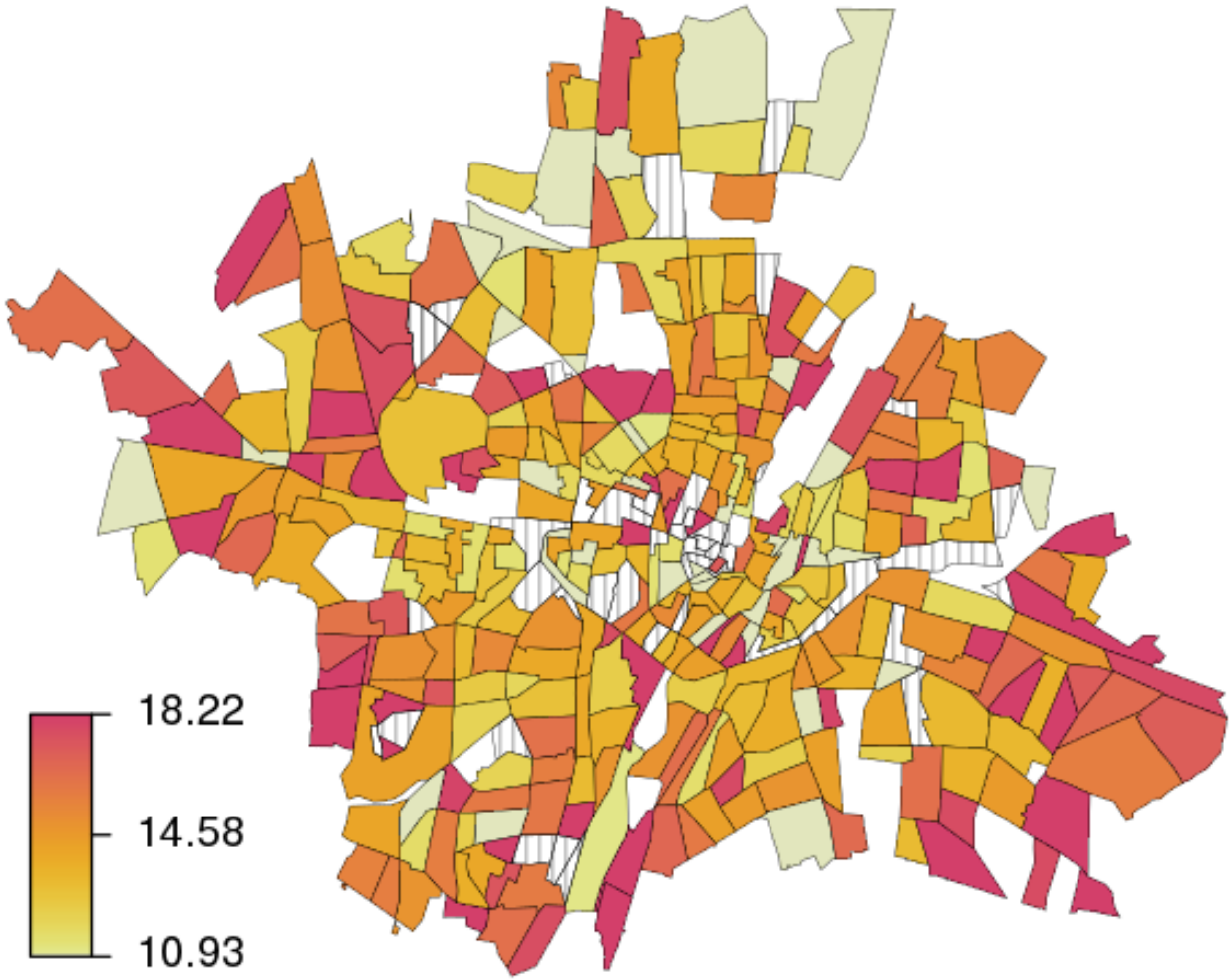
Spatial Covariates



Spatial Covariates

```
R> xr <- quantile(md$x, probs = c(0.1, 0.9))
R> xr
     10%      90%
10.93428 18.22036

R> data("MunichBnd", package = "bamlss")
R> plotmap(MunichBnd, x = md, col = heat_hcl,
+    symmetric = FALSE, swap = TRUE, range = xr)
```

Spatial Covariates

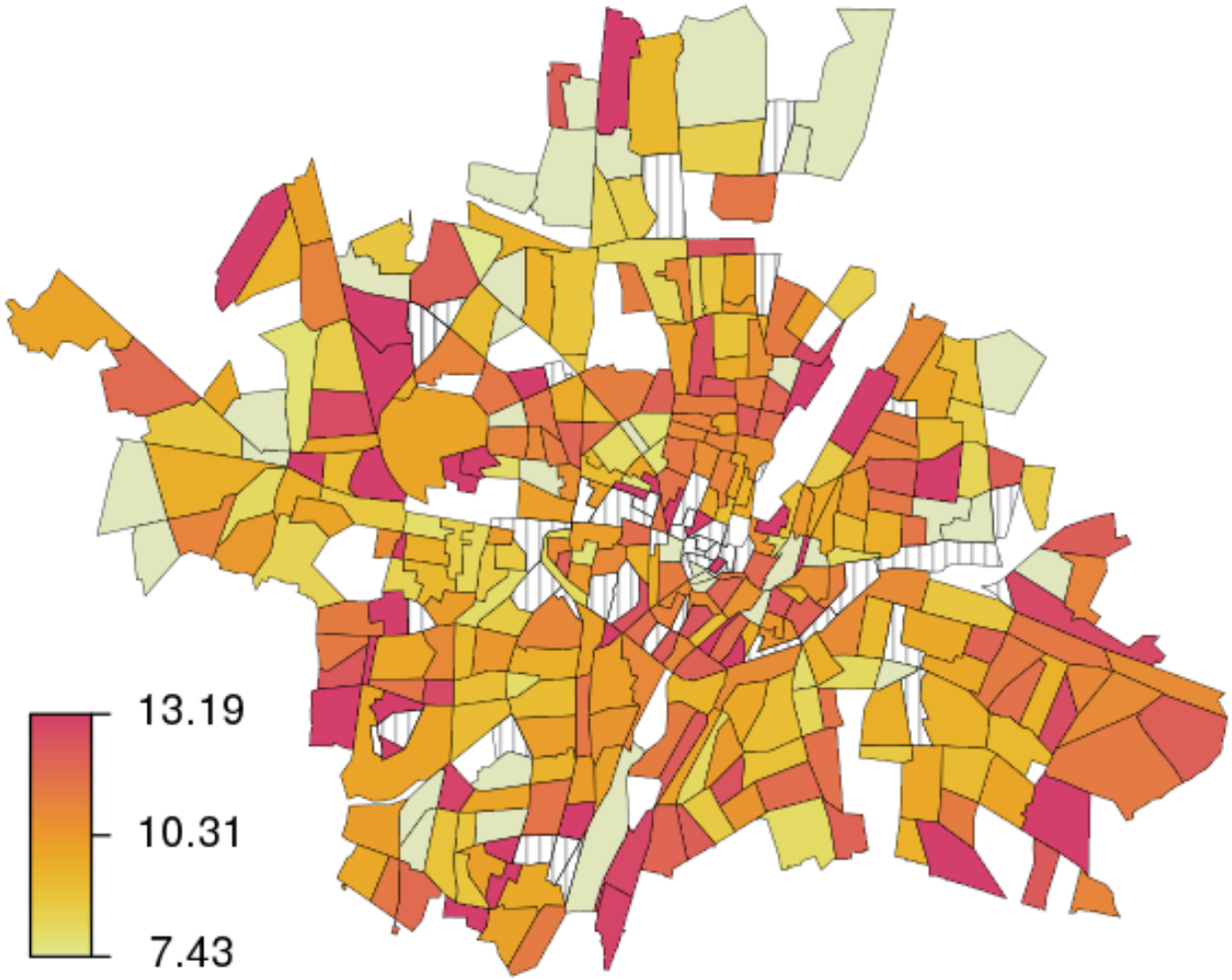


Spatial Covariates

Adding the district effect.

```
R> b3 <- lm(rentsqm ~ poly(area, 2) + poly(yearc, 2) +
+    bath + kitchen + cheating + location + as.factor(district),
+    data = rent99)

R> cb <- coef(b3)
R> nb <- names(cb)
R> i <- grep("district", nb)
R> csp <- cb[1] + cb[i]
R> id <- sapply(strsplit(nb[i], ")"), function(x) { x[2] })
R> xr <- quantile(csp, probs = c(0.1, 0.9))
R> xr

      10%       90%
 7.426949 13.186785

R> plotmap(MunichBnd, x = csp, id = id,
+    col = heat_hcl, symmetric = FALSE,
+    swap = TRUE, range = xr)
```

Spatial Covariates



More Regression Diagnostics

Regression Diagnostics

- The two basic components of many diagnostics are the *fitted values* and the *residuals*.
- The ith fitted value is the estimate of $E(y_i)$ from the model,

$$\hat{y}_i = \mathbf{x}_i^\top \hat{\boldsymbol{\beta}},$$

  where $\mathbf{x}_i$ is the ith row of the design matrix $\mathbf{X}$.
- The vector of all n fitted values is

$$\hat{\mathbf{y}} = \mathbf{H}\mathbf{y},$$

  where

$$\mathbf{H} = \mathbf{X}(\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top.$$

- The ith residual is defined to be

$$\varepsilon_i = y_i - \hat{y}_i.$$

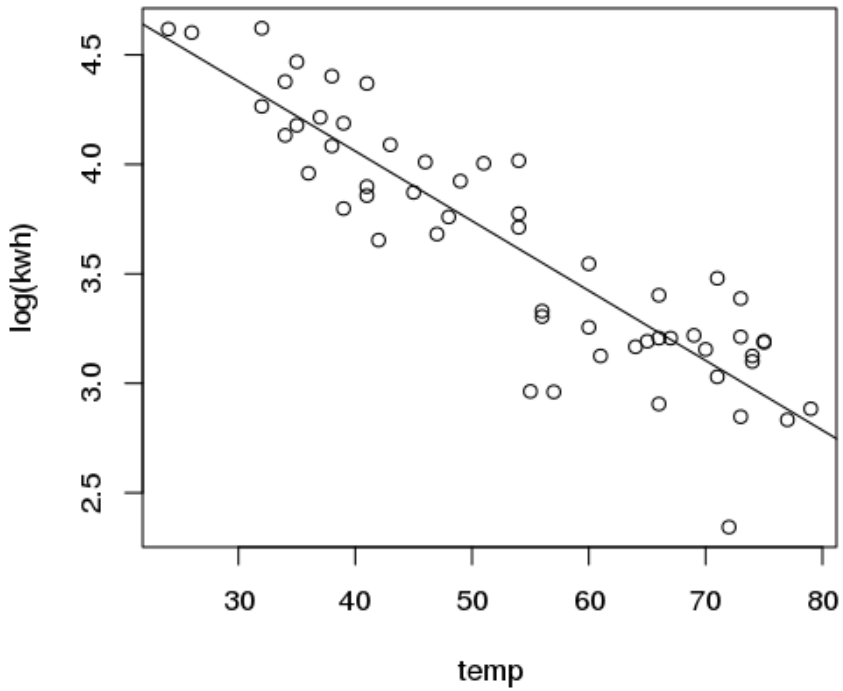
Regression Diagnostics

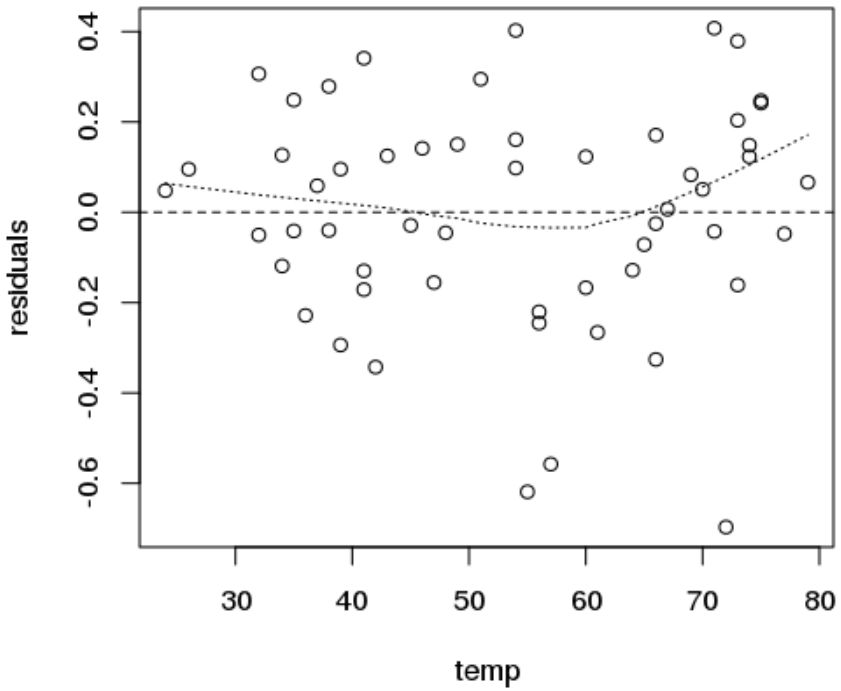
Properties of the hat matrix:

1. $\mathbf{H}$ is symmetric.
2. $\mathbf{H}$ is idempotent, $\mathbf{H} = \mathbf{H}^\top = \mathbf{H}^2$.
3. $rank(\mathbf{H}) = trace(\mathbf{H}) = p$.
4. $\frac{1}{n} \leq H_{ii} \leq 1$.
5. $\mathbf{I} - \mathbf{H}$ is symmetric and idempotent, $rank(\mathbf{I} - \mathbf{H}) = n - p$.

Regression Diagnostics

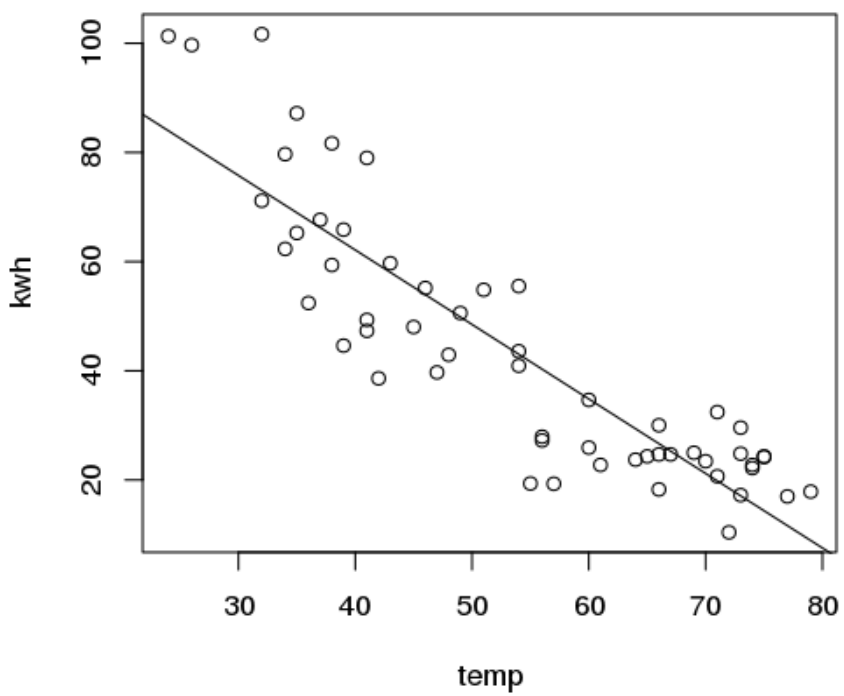
- Any patterns in the residuals reflect extra structure that is not accommodated by the model.
- Example, electricity usage study, $log(\texttt{kwh}) = \beta_0 + \beta_1 \texttt{temp}$.

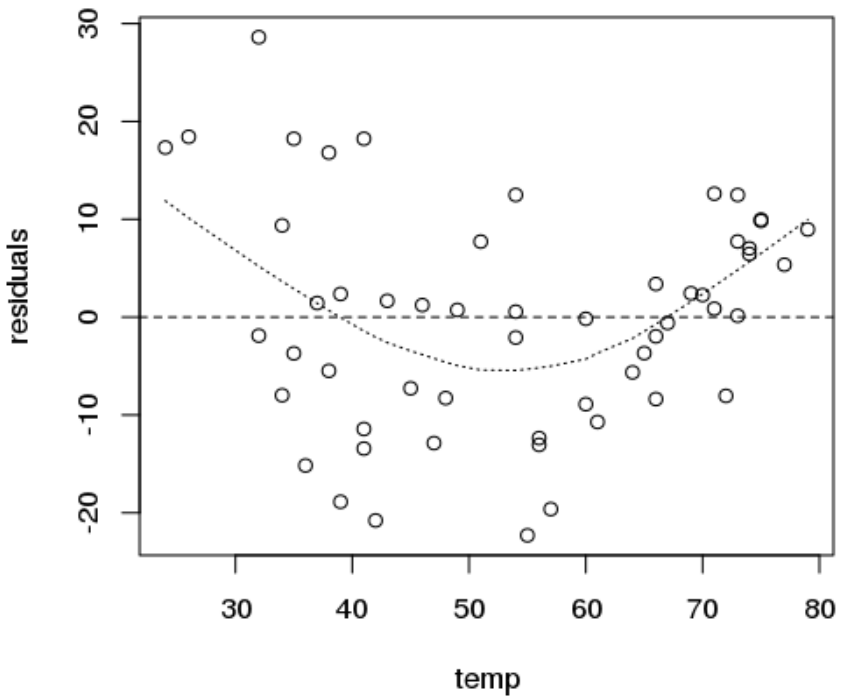




Regression Diagnostics

- The points are scattered around the zero line without any strong patterns. The simple log-linear model is reasonable in this case.
- If there had been a curvelinear pattern to the residuals, this would have been evidence of lack of fit of the simple linear model.
- Example, electricity usage study, $\texttt{kwh} = \beta_0 + \beta_1 \texttt{temp}$.





Regression Diagnostics

Normalized Residuals

- Technique in the detection of outliers.
- The vector of residuals is

$$\boldsymbol{\varepsilon} = \mathbf{y} - \hat{\mathbf{y}} = \mathbf{y} - \mathbf{H}\mathbf{y} = (\mathbf{I} - \mathbf{H})\mathbf{y},$$

  where

$$Cov(\boldsymbol{\varepsilon}) = Cov((\mathbf{I} - \mathbf{H})\mathbf{y}) = (\mathbf{I} - \mathbf{H})\sigma^2(\mathbf{I} - \mathbf{H})^\top = \sigma^2(\mathbf{I} - \mathbf{H}).$$

- $$\widehat{\text{st.dev.}}(\varepsilon_i) = \hat{\sigma}\sqrt{1 - H_{ii}}.$$

- The ith *internally studentized (normalized) residual* is

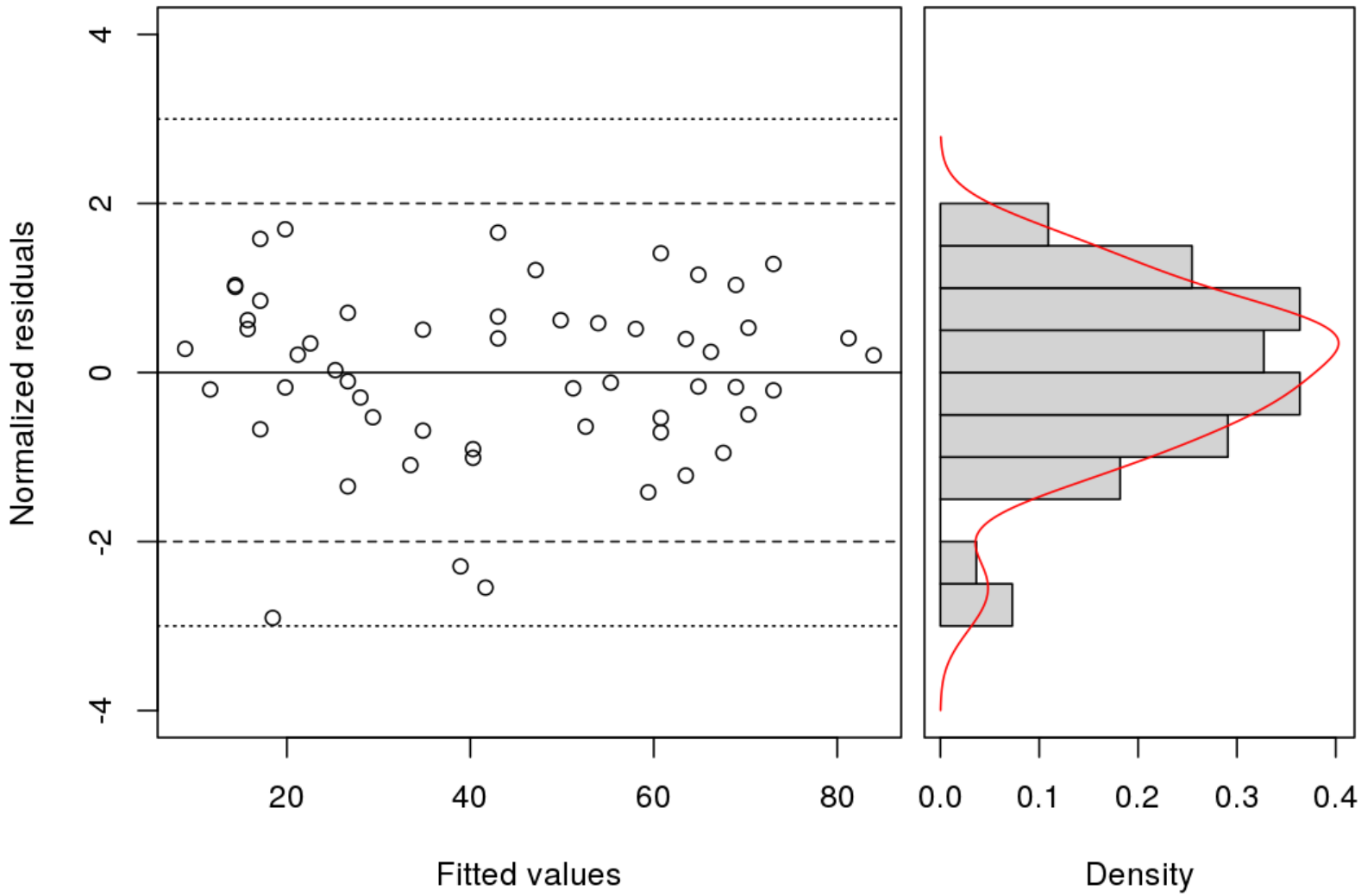
$$\varepsilon_i^\star \equiv \frac{\varepsilon_i}{\widehat{\text{st.dev.}}(\varepsilon_i)} = \frac{\varepsilon_i}{\hat{\sigma}\sqrt{1 - H_{ii}}}.$$

Regression Diagnostics

Under normality and homoscedasticity $\varepsilon_i^\star$ should behave like a $N(0, 1)$ sample if the model is correct.

```
R> library("SemiPar")
R> data("elec.temp")
R> attach(elec.temp)

R> n <- length(temp)
R> X <- cbind(1, temp)
R> H <- X %*% solve(t(X) %*% X) %*% t(X)
R> e <- (diag(n) - H) %*% log(usage)
R> p <- sum(diag(H))
R> sigma <- sqrt(t(e) %*% e * 1 / (n - p))
R> e.norm <- e / (sigma * sqrt(1 - diag(H)))

R> plot(H %*% log(usage), e.norm, ylim = c(-4, 4),
+    xlab = "Fitted values",
+    ylab = "Normalized residuals")
R> abline(h = c(-3, -2, 0, 2, 3), lty = c(3, 2, 1, 2, 3))
```

Regression Diagnostics



Regression Diagnostics

- The residuals should be evenly scattered above and below zero.
- A trend in the mean would violate the assumption of independent response variables, usually results from an erroneous model structure.
- A trend in the variability of the residuals suggests that the variance of the response is related to its mean, violating the constant variance assumption.
- Transformation of the response or a *GLM* may help.

Regression Diagnostics

Quantile-Quantile Plots

- The cumulative distribution function of a random variable X is

$$F(x) = P(X \leq x) = p.$$

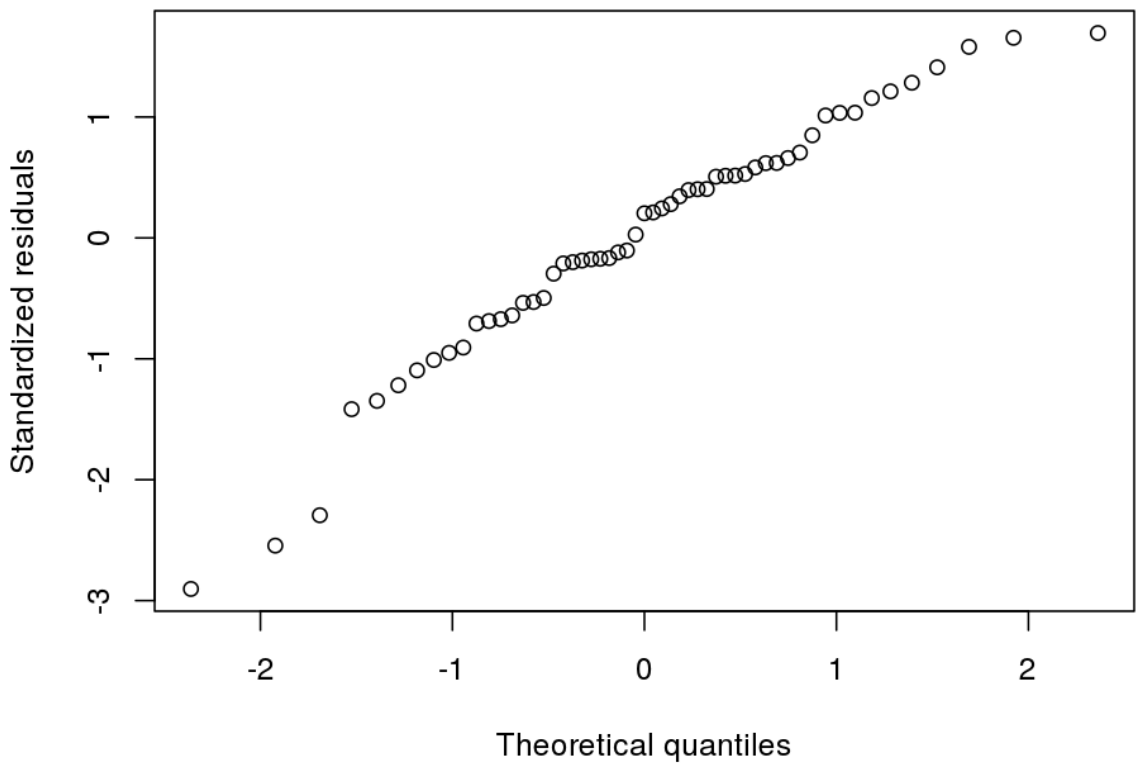
  The inverse Q is called the quantile function.
- If F is a one to one function the inverse Q is uniquely determined

$$Q(p) = x \text{ if } F(x) = p.$$

- The idea is to plot the theoretical quantiles against the quantiles we generated from the residuals.
- If the residuals are normally distributed then the resulting plot should look like a straight line relationship.

Regression Diagnostics

```
R> e.norm <- sort(e.norm)
R> t.probs <- (1:n - 0.5) / n
R> t.quantiles <- qnorm(t.probs, mean = 0, sd = 1)
R> plot(t.quantiles, e.norm,
+    xlab = "Theoretical quantiles",
+    ylab = "Standardized residuals")
```



Regression Diagnostics

Hat Diagonals or Leverages

- How much potential do outliers have on the fitted line?
- The ith leverage value is the ith diagonal of the hat matrix, H_{ii}.
- From $\hat{\mathbf{y}} = \mathbf{H}\mathbf{y}$ we get

$$\hat{y}_i = \sum_{j=1}^{n} H_{ij} y_j = H_{i1} y_1 + \ldots + H_{ii} y_i + \ldots H_{in} y_n.$$

- H_{ii} is the weight of y_i, i.e. the influence of y_i on its own fitted value.
- H_{ii} only depents on the predictors, H_{ii} measures only the potential for being influential and not actual influence.
- Example; in linear regression with a single predictor x, H_{ii} is a linear function of the squared distance from x_i to $\bar{x}$, a measure how "outlying" x_i is, the larger H_{ii} the more outlying is x_i.

Regression Diagnostics

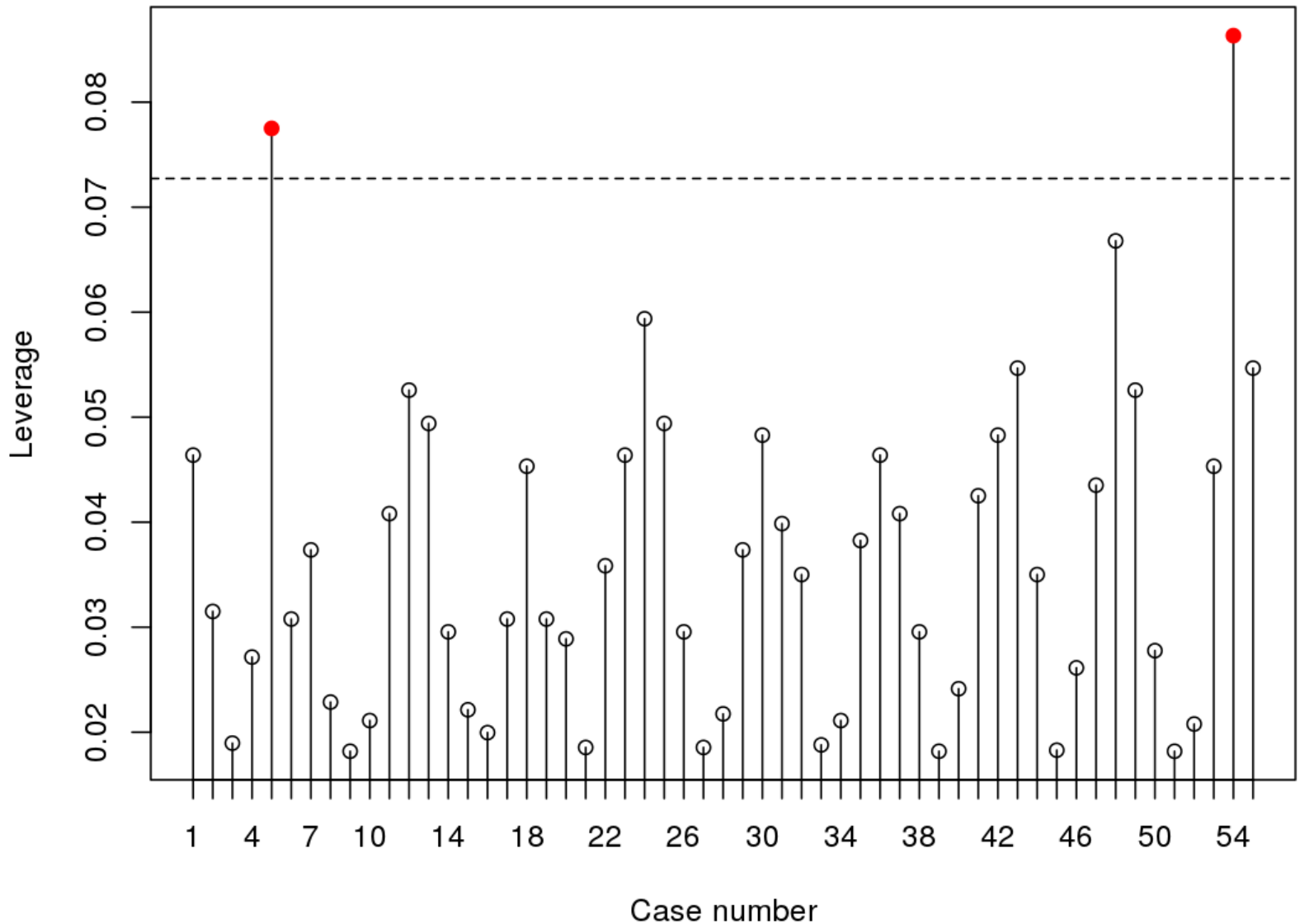
- For large values of H_{ii}, $\sigma\sqrt{1 - H_{ii}}$ is small.
- If H_{ii} is close to 1, $(y_i, \mathbf{x}_i)$ almost surely lies on the regression line (plane).

```
R> plot(diag(H), type = "h", xlab = "Case number",
+    ylab = "Leverage", axes = FALSE)
R> axis(1, at = 1:n, labels = 1:n); axis(2); box()
R> col <- rep("black", n)
R> bg.col <- rep(NA, n)
R> col[diag(H) >= 2 * p/n] <- "red"
R> bg.col[diag(H) >= 2 * p/n] <- "red"
R> points(1:n, diag(H), pch = 21, col = col, bg = bg.col)
R> abline(h = 2 * p/n, lty = 2)
```

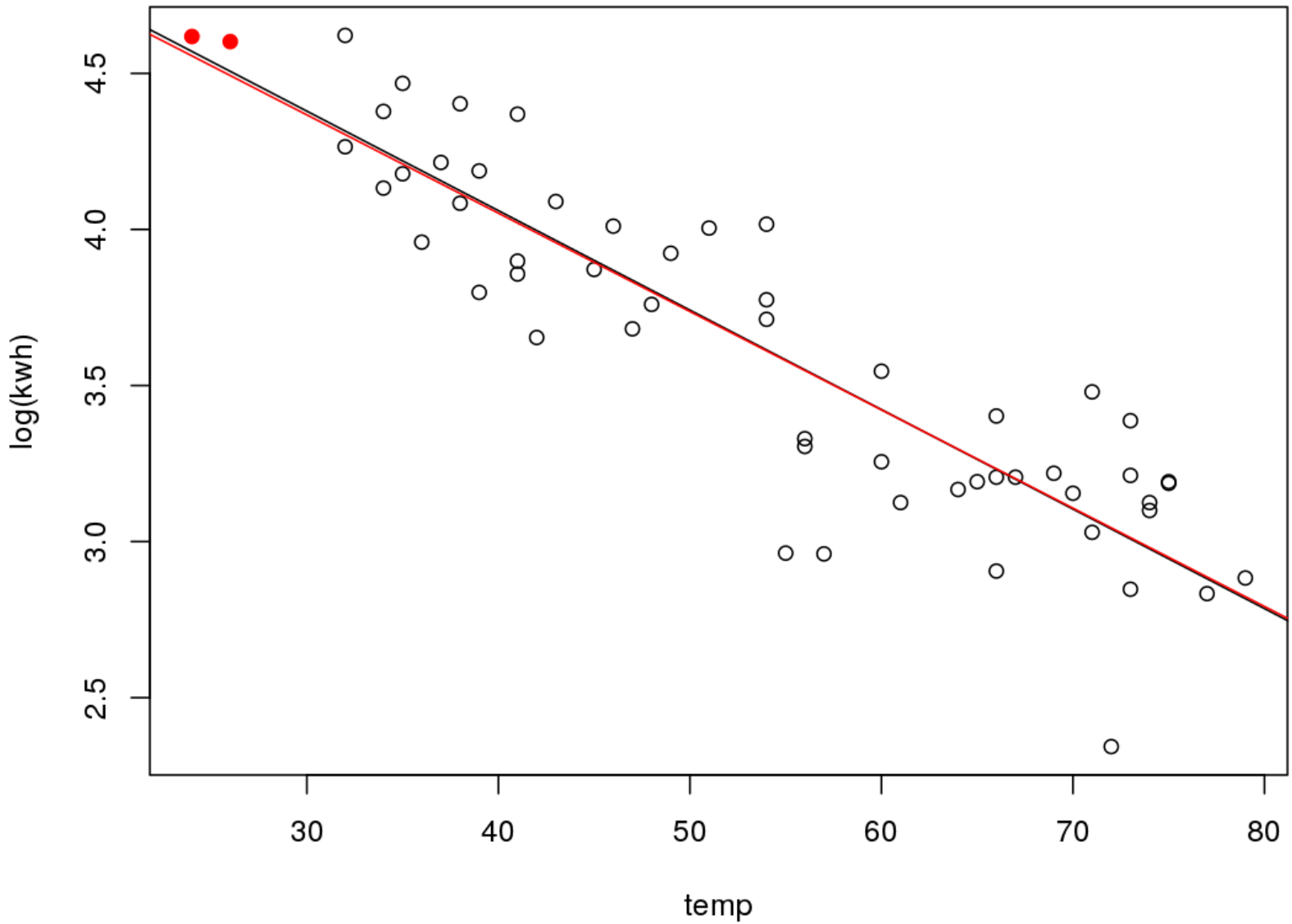
```
R> (1:n)[diag(H) >= 2 * p/n]
```

```
[1]  5 54
```

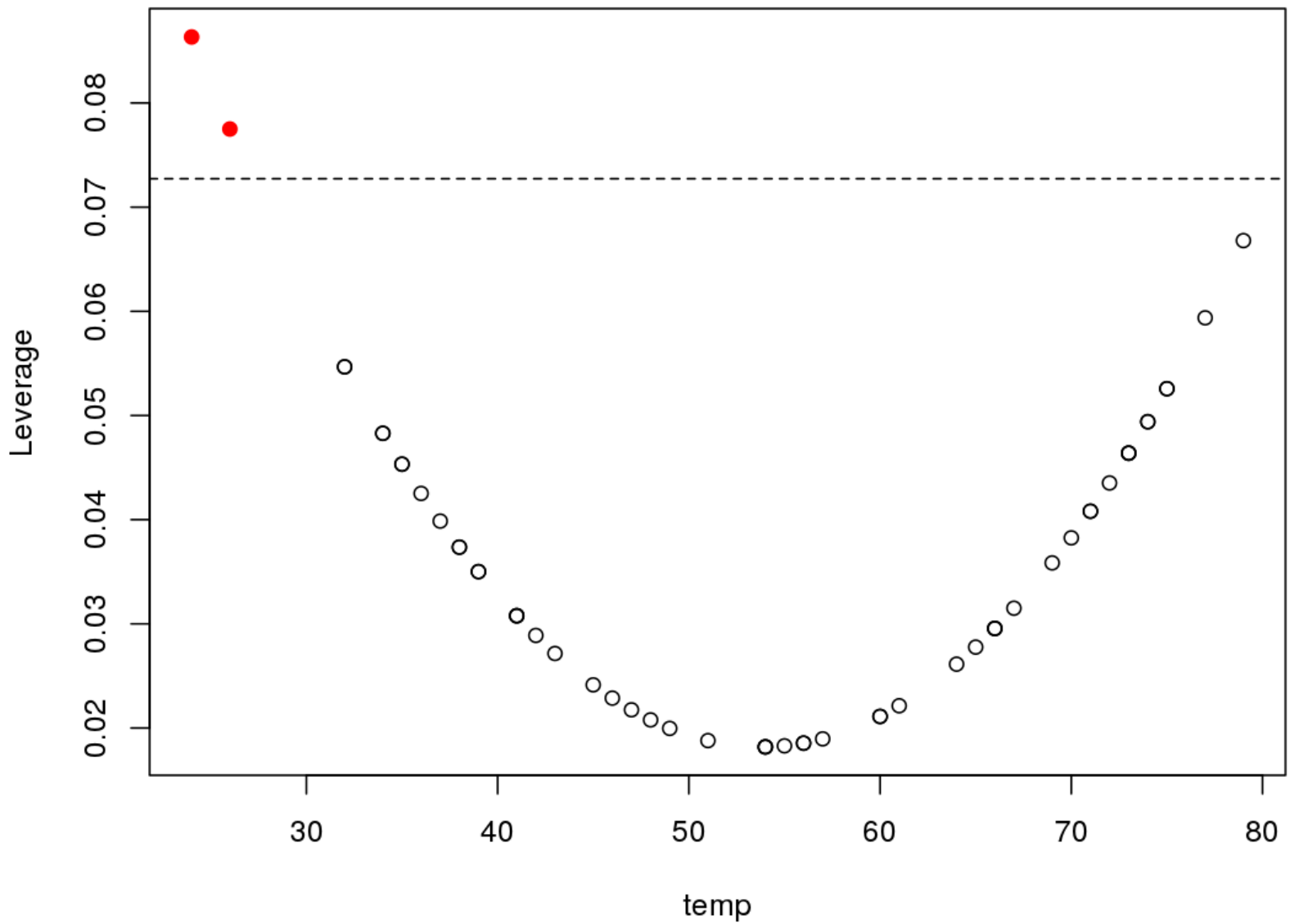
Regression Diagnostics



Regression Diagnostics



Regression Diagnostics



Regression Diagnostics

Cook's Distance

- Another way to measure influence of certain data points on the regression line.
- Let $\hat{\mathbf{y}}$ and $\hat{\mathbf{y}}^{(k)}$ be, respectively, the vector of fitted values using all the data and with the kth case deleted.
- Cook's Distance is defined

$$D_i = \frac{||\hat{\mathbf{y}} - \hat{\mathbf{y}}^{(k)}||^2}{p\hat{\sigma}^2} = \frac{(\hat{\mathbf{y}} - \hat{\mathbf{y}}^{(k)})^\top(\hat{\mathbf{y}} - \hat{\mathbf{y}}^{(k)})}{p\hat{\sigma}^2}.$$

- It can be shown that

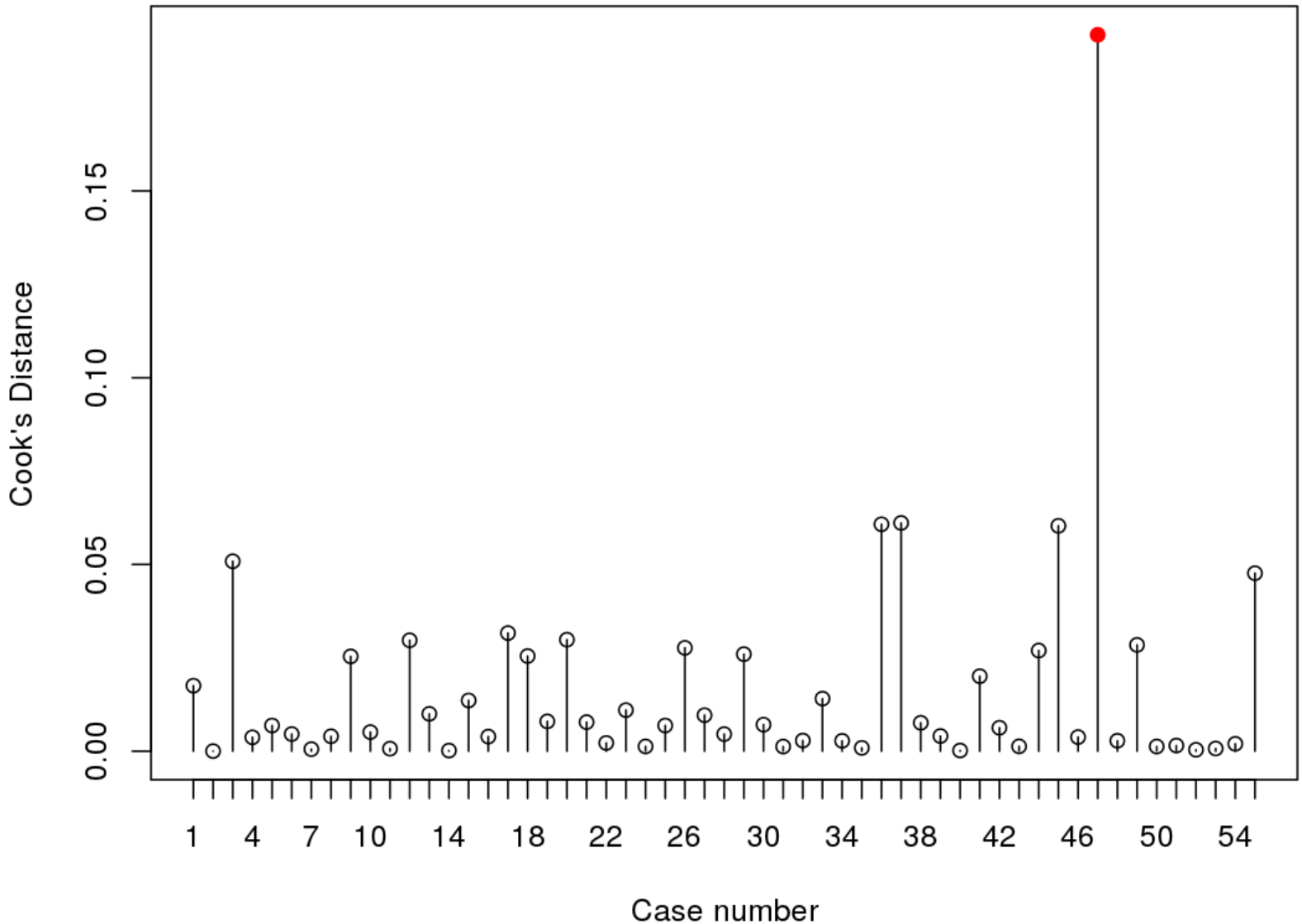
$$D_i = \frac{(\varepsilon_i^\star)^2 H_{ii}}{p(1 - H_{ii})}.$$

- Values of $D_i > 0.5$ are considered to be conspicuous.

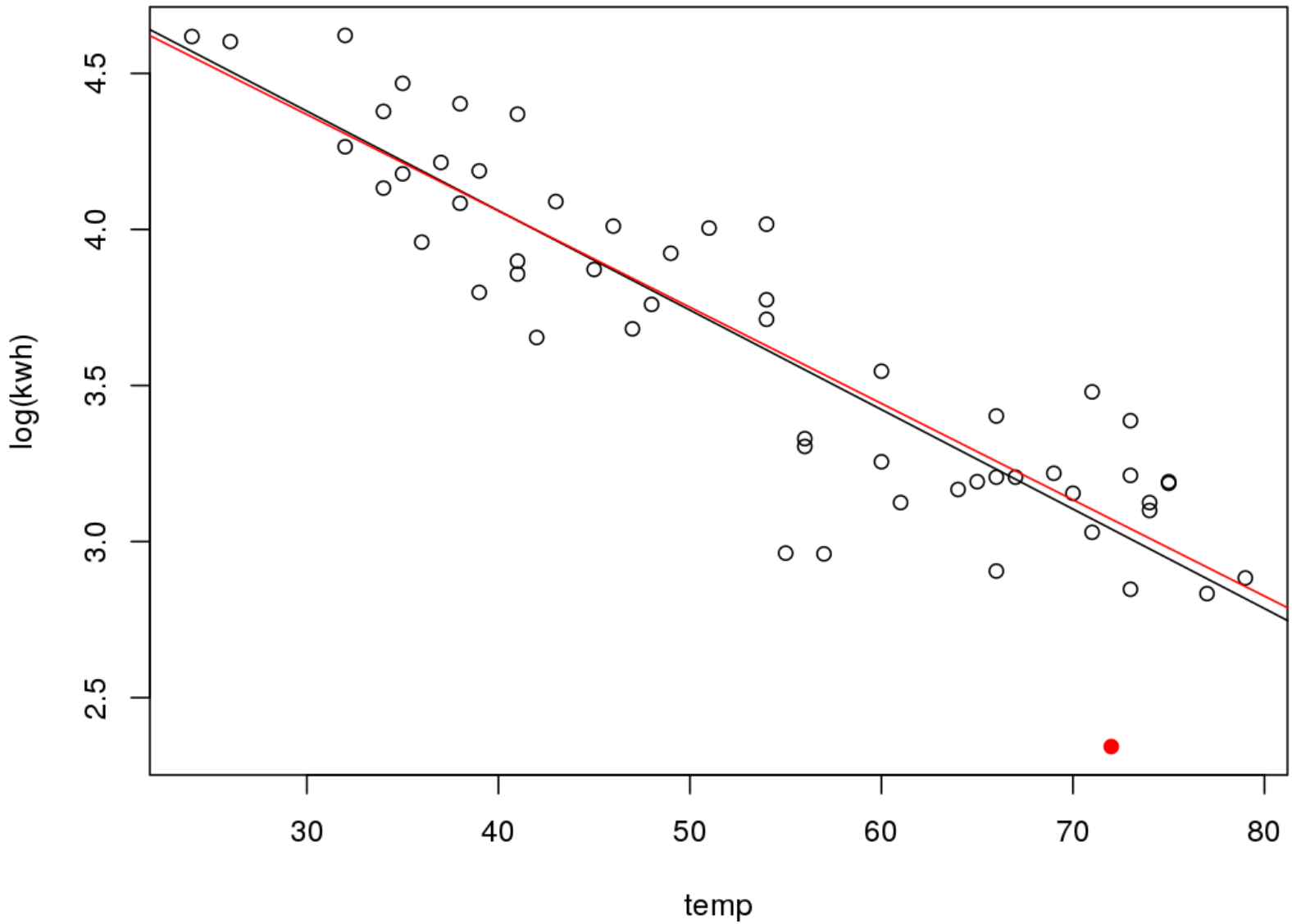
Regression Diagnostics

```
R> yhat <- H %*% log(usage)
R> ind <- 1:n
R> D <- NULL

R> for(k in 1:n) {
+    Xk <- X[ind != k, ]
+    beta <- solve(t(Xk) %*% Xk) %*% t(Xk) %*% log(usage)[ind != k]
+    yhatk <- X %*% beta
+    D <- c(D, t((yhat - yhatk)) %*% (yhat - yhatk) / (p * sigma^2))
+  }

R> plot(D, type = "h", axes = FALSE,
+    xlab = "Case number", ylab = "Cook's Distance")
R> axis(1, at = 1:n); axis(2); box()
R> col <- rep("black", n); bg.col <- rep(NA, n)
R> col[D == max(D)] <- "red"; bg.col[D == max(D)] <- "red"
R> points(1:n, D, pch = 21, col = col, bg = bg.col)

R> c((1:n)[D == max(D)], max(D))

[1] 47.0000000  0.1918169
```

Regression Diagnostics



Regression Diagnostics



Regression Diagnostics

Autocorrelation

- An important assumption in regression analysis is that the errors are independent or, at least, uncorrelated.
- If this assumption is false, then least squares estimation could be inefficient.
- More seriously, commonly used inferential procedures are invalid when there is autocorrelation.
- Denote $\hat{\rho}(k)$ as the sample correlation between ε_i and ε_{i-k} with $k+1 \leq i \leq n$.
- For general $k = 0, 1, \ldots, n-1$;

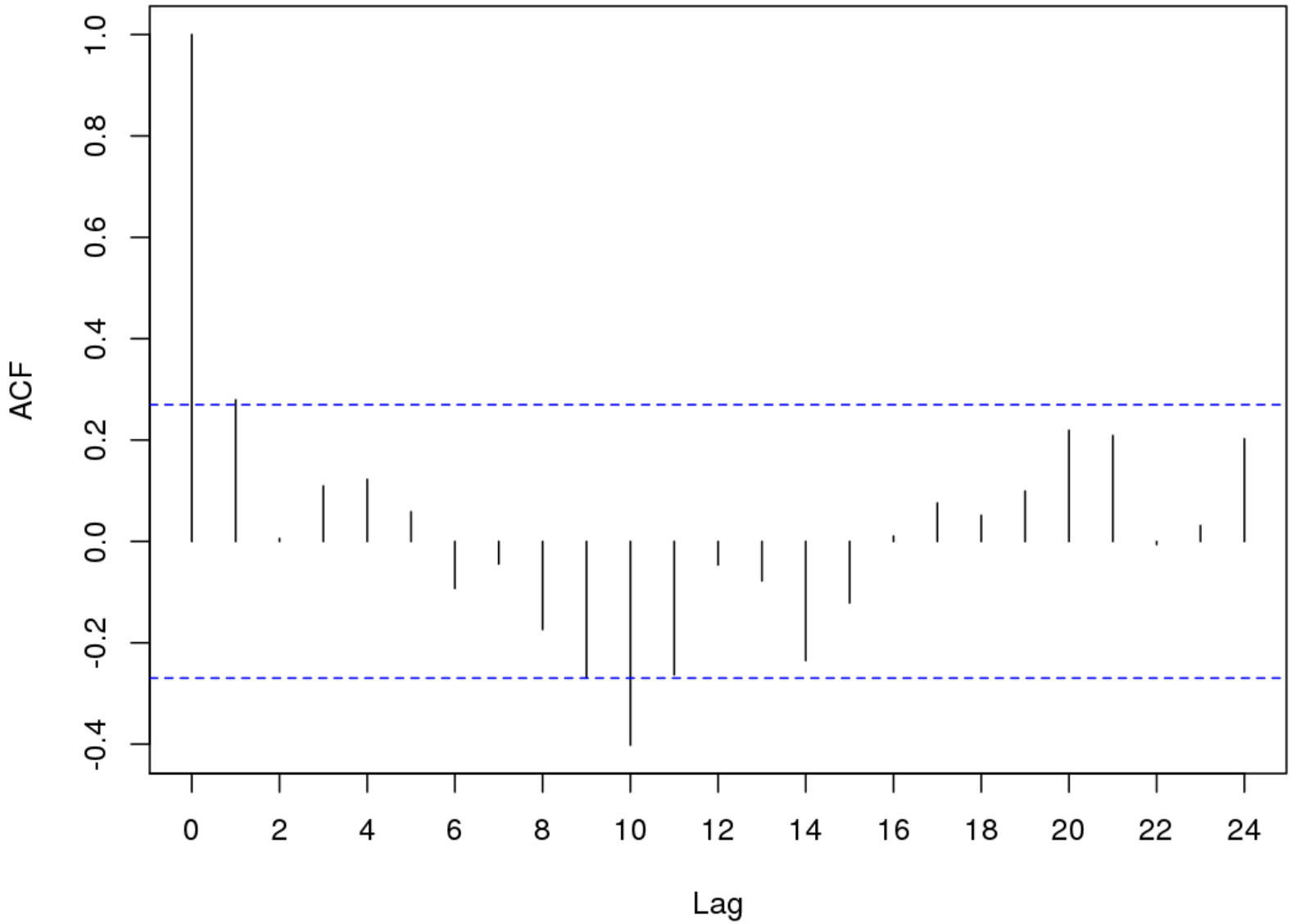
$$\hat{\rho}(k) = \frac{\sum_{i=k+1}^{n} \varepsilon_i \varepsilon_{i-k}}{\sum_{i=1}^{n} \varepsilon_i^2}.$$

Regression Diagnostics

- If the errors are independent, $\hat{\rho}(k)$ is approximately $N(0, 1/n)$, thus, any value of $|\hat{\rho}(k)| > 2/\sqrt{n}$ is roughly significant (on a 95% level).

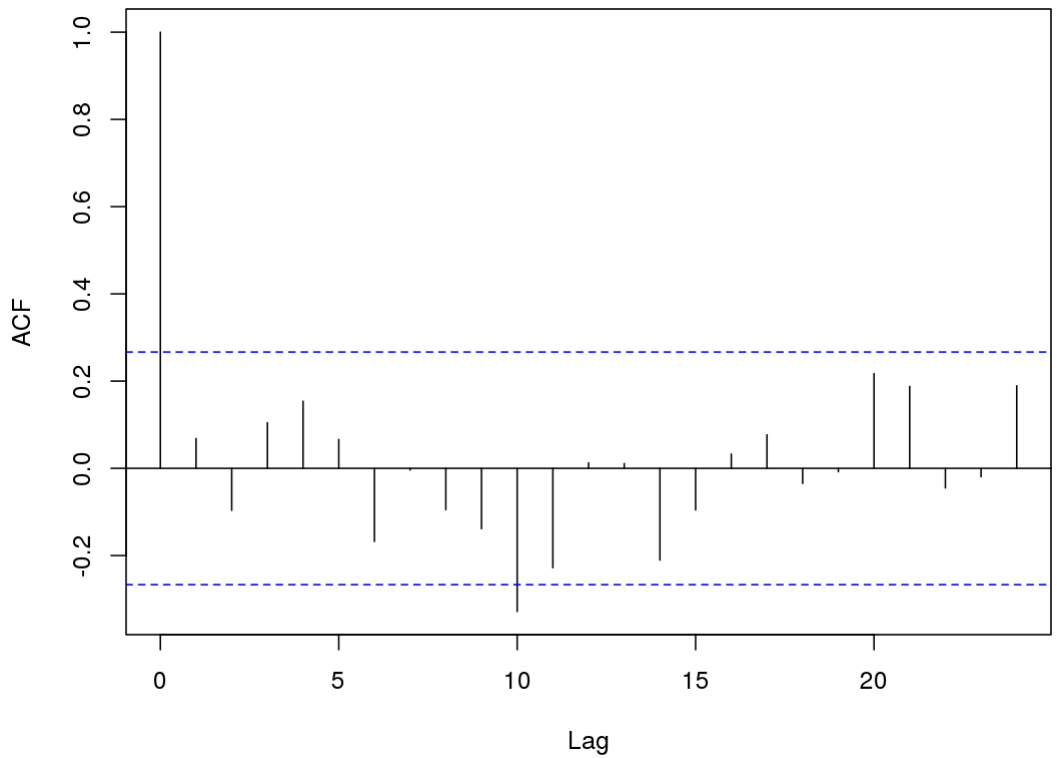
```
R> e2 <- sum(e^2)
R> rho <- NULL

R> for(k in 0:24) {
+    j <- k + 1
+    rho <- c(rho, sum(e[j:n] * e[j:n - k]) / e2)
+  }

R> plot(rho, type = "h", xlab = "Lag", ylab = "ACF", axes = FALSE)
R> at <- pretty(0:length(rho), n = 10) - 1
R> axis(1, at = at, labels = at - 1)
R> axis(2); box()
R> abline(h = c(-2/sqrt(n), 2/sqrt(n)), lty = 2, col = "blue")
```

Regression Diagnostics



Regression Diagnostics

```
R> X.l <- cbind(1, temp, c(NA, head(usage, -1)),
+    c(NA, head(temp, -1)))
R> X.l <- na.omit(X.l)
R> H.l <- X.l %*% solve(t(X.l) %*% X.l) %*% t(X.l)
R> e.l <- log(usage[2:n]) - H.l %*% log(usage[2:n])
```



Regression Diagnostics

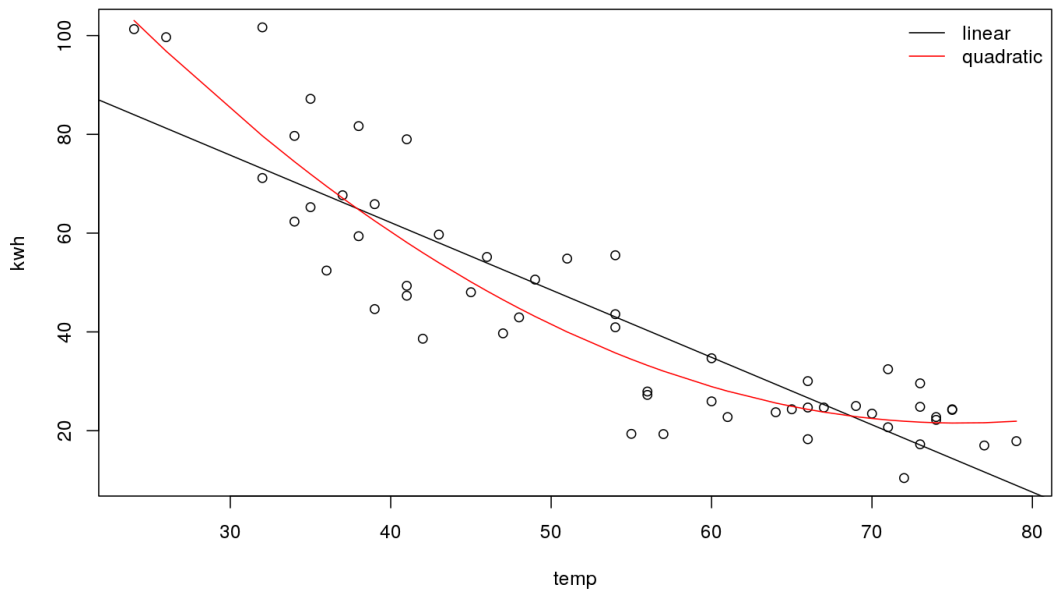
Model Selection

Model Selection

Extra Sums of Squares

- We often need to compare two or more models.
- E.g., if one uses `kwh` as the response variable then there is evidence of lack of fit to the straight line model.
- Now consider a quadratic model

$$\texttt{kwh}_i = \beta_0 + \beta_1 \texttt{temp}_i + \beta_2 \texttt{temp}_i^2 + \varepsilon_i.$$



Model Selection

- The quadratic fit appears to be better, but is this just due to random variation or is there really a curvature that requires an alternative to the straight line model?
- We can address this question by testing

$$H_0 : \beta_2 = 0 \text{ versus } H_1 : \beta_2 \neq 0.$$

- The lest squares estimator minimizes $RSS = \varepsilon^\top \varepsilon$, which is a cursory measure of the quality of the fit. If one model fits better than the other, then this difference in fit should be evident in the RSS values.
- The extra sum of squares is $ExtraSS \equiv RSS_{\text{linear}} - RSS_{\text{quadratic}}$, or more general

$$ExtraSS \equiv RSS_{\text{smaller}} - RSS_{\text{larger}}.$$

Model Selection

- The extra sum of squares can be used to test the larger versus the smaller models.
- The null hypothesis is that the smaller models fit the data.
- Let p_{smaller} and p_{larger} be the number of parameters of the respective models. The F-test statistic is

$$F \equiv \frac{ExtraSS/(p_{\text{larger}} - p_{\text{smaller}})}{\hat{\sigma}^2_{\text{larger}}}.$$

- Under the null hypothesis and the normality assumption, F has a F-distribution with $p_{\text{larger}} - p_{\text{smaller}}$ and $n - p_{\text{larger}}$ degrees of freedom.
- The F-statistic compares the improvement per parameter to the data variability. The null hypothesis should be rejected when F is large.

Model Selection

- An α-level test is obtained by rejecting the null hypothesis when F exceeds the $1 - \alpha$ quantile of the F-distribution with $p_{\text{larger}} - p_{\text{smaller}}$ and $n - p_{\text{larger}}$ degrees of freedom.

```
R> b1 <- lm(usage ~ temp); b2 <- lm(usage ~ temp + I(temp^2))
R> e1 <- residuals(b1); e2 <- residuals(b2)
R> RSS1 <- t(e1) %*% e1; RSS2 <- t(e2) %*% e2;
R> p1 <- length(coef(b1)); p2 <- length(coef(b2))
R> sigma2 <- drop(RSS2/b2$df.residual)
R> F <- drop(((RSS1 - RSS2) / (p2 - p1)) / sigma2)
R> F

[1] 22.55071

R> pvalue <- 1 - pf(F, p2 - p1, n - p2)
R> pvalue

[1] 0.00001647759
```

Model Selection

- Hypothesis testing is “biased” towards the null hypothesis in that we retain the null unless there is strong evidence to the contrary.
- When we want to compare two models, we often want to treat them on an equal footing rather than accepting one unless there is strong evidence against it.
- One alternative is to try to find the model that gets as close as possible to the true model.
- We can attempt to find the model which does the best job of predicting $E(y_i)$.

Model Selection

Cross Validation (CV)

- The *RSS* of a model is a measure of predictive ability, however, it is not satisfactory as a model selector.
- The problem is that the model with the largest number of parameters and containing the other models as special cases always has the smallest *RSS*.
- Now consider the model $y_i = f(x_{i1}, \ldots, x_{ik}) + \varepsilon_i$. Ideally, it would be good to choose a model so that the estimate $\hat{f}$ is as close as possible to f.
- A suitable criterion might be

$$M = \frac{1}{n} \sum_{i=1}^{n} \left(\hat{f}_i - f_i \right)^2 .$$

Model Selection

- Since f is unknown, M cannot be computed directly, but it is possible to derive an estimate of $E(M) + \sigma^2$.
- Let $\hat{f}^{(k)}$ be the model fitted to all data except y_k, the *ordinary cross validation* score is

$$V_o = \frac{1}{n} \sum_{i=1}^{n} \left(\hat{f}_i^{(k)} - y_i \right)^2 .$$

- Substituting $y_i = f_i + \varepsilon_i$,

$$\begin{aligned} V_o &= \frac{1}{n} \sum_{i=1}^{n} \left(\hat{f}_i^{(k)} - f_i - \varepsilon_i \right)^2 \\ &= \frac{1}{n} \sum_{i=1}^{n} \left(\hat{f}_i^{(k)} - f_i \right)^2 - \left(\hat{f}_i^{(k)} - f_i \right) \varepsilon_i + \varepsilon_i^2 . \end{aligned}$$

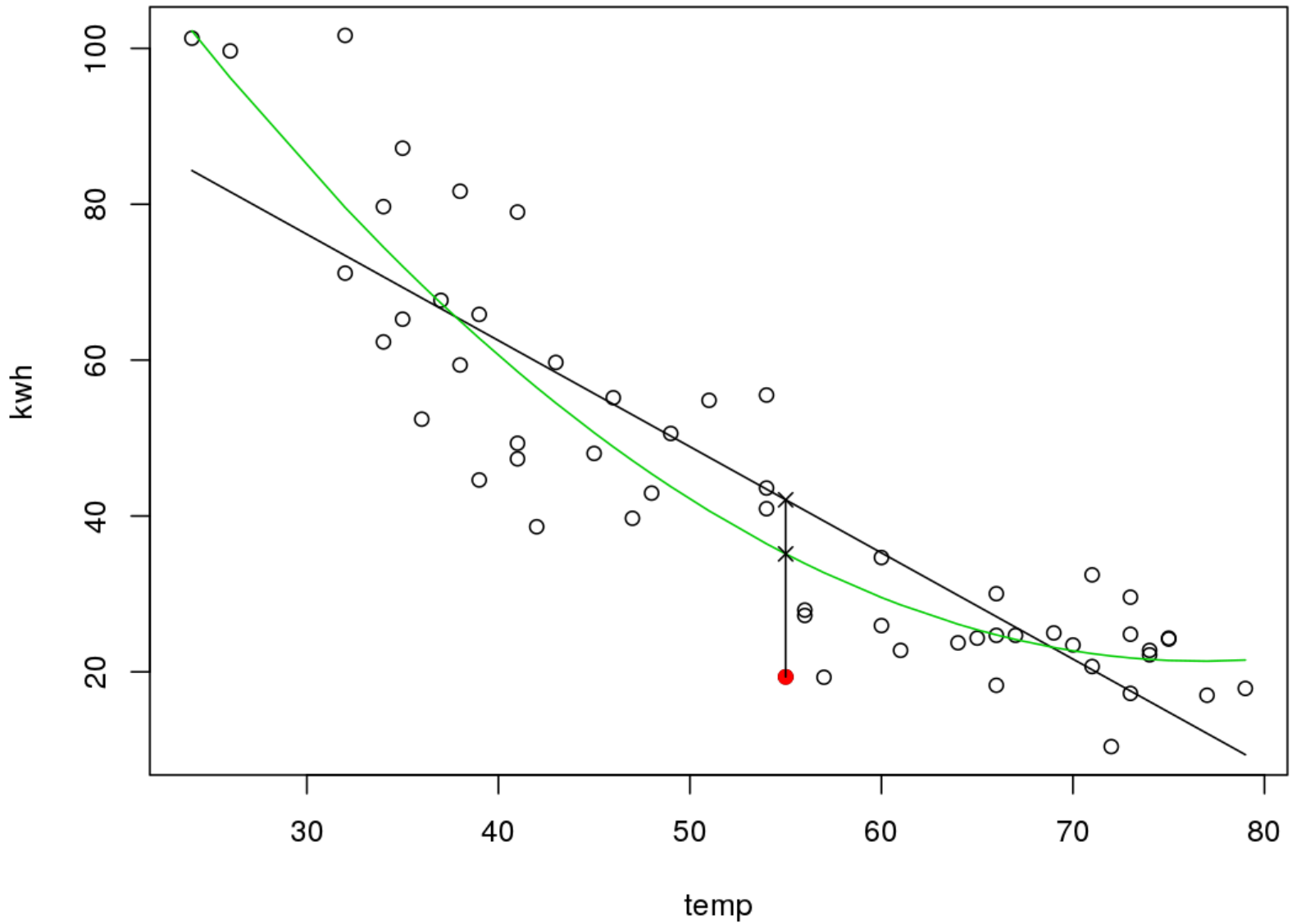
Model Selection

- Since $E(\varepsilon_i) = 0$ and ε_i and $\hat{f}^{(k)}$ are independent, we get

$$E(V_o) = \frac{1}{n} E\left(\sum_{i=1}^{n} \left(\hat{f}_i^{(k)} - f_i\right)^2\right) + \sigma^2.$$

- Now, in the large limit, $\hat{f}^{(k)} \approx \hat{f}$, so $E(V_o) \approx E(M) + \sigma^2$, hence choosing the model that minimizes V_o is a reasonable approach if the ideal would be to minimize M.

Model Selection



Model Selection

```
R> cv(usage, temp)
[1] 134.7991
R> cv(usage, temp, temp^2)
[1] 95.41717
```

Model Selection

Akaike Information Criterion (AIC)

- Another popular model selection criterion is the AIC.
- The criterion is based on the *maximum likelihood* (ML) of the conditional probability density function of the response. In our case the likelihood is

$$L(\mathbf{y}|\mathbf{X}, \boldsymbol{\beta}, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^\top(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})}{2\sigma^2}\right)$$

  The *log-likehood* is

$$log(L) = -\frac{n}{2}log(2\pi) - \frac{n}{2}log(\sigma^2) - \frac{1}{2\sigma^2}\boldsymbol{\varepsilon}^\top\boldsymbol{\varepsilon}.$$

- Using ML would intuitively suggest to select the model with the largest *(log-)likelihood*

Model Selection

- Again, if model choice is based on ML, we would hence estimate models with an unnecessary large number of parameters.
- Now, suppose that the estimated parameters of a model are $\hat{\boldsymbol{\theta}}^\top = (\hat{\theta}_1, \ldots, \hat{\theta}_p)$, to correct for the bias the AIC is defined by

$$AIC = -2log(L) + 2p.$$

- Comparing two (or more) models, the difference of AIC values is considered to be "significant" if

$$|AIC(M_1) - AIC(M_2)| > 2.$$

Model Selection

```
R> aic(usage, temp)
[1] 427.3915
R> aic(usage, temp, temp^2)
[1] 409.6466
```

Model Selection

Bayesian Information Criterion (BIC)

- A criterion that penalizes model complexity even more is the BIC.
- The BIC is defined by

$$BIC = -2log(L) + p \cdot log(n).$$

- For $log(n) \geq 2 \Leftrightarrow n \geq 8$, the BIC clearly penalizes model complexity more than the AIC.

Summary of Available Functions in R

Luckily, most of the described diagnostics and model selection tools are already implemented in R.

`lm()`	fits linear regression models
`summary()`	standard regression output
`coefficients()`	extracting the regression coefficients
`residuals()`	extracting residuals
`fitted()`	extracting fitted values
`anova()`	comparison of nested models
`predict()`	prediction for new data
`plot()`	diagnostic plots
`confint()`	confidence intervals for the regression coefficients
`deviance()`	residual sum of squares
`vcov()`	variance-covariace matrix
`logLik()`	log-likelihood
`AIC()`, `BIC()`	information criteria

Summary

- Nowadays, regression problems consistently grow in complexity.
- One problem all statistical models have in common is the question of how appropriate the relationships between variables can be examined?
- In a number of cases the classical regression model is not sufficient enough to model highly complex relationships, e.g., in cases where the shape of a functional relationship is not predetermined ordinary regression models do not produce satisfying fits.
- Transformation of covariates/response may improve the model fit but do not suffice in a number of cases.
- Dummy regression for spatial data seems not to account well for the inherent neighborhood structure of such data.
- ...