Data Analysis Learning

关于 CWorld 学习 Analysis Learning 一些笔记和代码。该课程使用 R 语言进行数据分析。

Get started

Hint

点击侧栏的目录或下滑以阅览更多章节。 当然，你也可以下载 PDF 版本 的笔记。

Development

如果你对该项目有兴趣，请前往 Github 了解更多。

Contributions

由于作者只是个正在浅学 Database 的初学者，所以笔记难免存在明显纰漏，还请读者们多多海涵。此外，也欢迎诸位使用 PR 或 Issues 来改善它们。

Thanks

一些电子教材对作者学习上帮助颇多，没有这些资料，就没有这部笔记。在此对这些教材的原作者深表感谢。读者若对此项目笔记抱有疑惑，也可以仔细阅读以下教材以作弥补。

- STATS 201 : Data Analysis

Table of Contents

Skip to main content

At the beginning

章节

- Chapter1: Getting started with regression
- Chapter2: Basics of simple linear regression
- Chapter3: The null model
- Chapter4: Dealing with Curves
- Chapter5: Dealing with fact or data with two levels
- Chapter6: Dealing with multiplicative relationships
- Chapter7: Dealing with power relationships
- Chapter8: Dealing with numerical and fact or explanatory variables - part 1
- Chapter9: Dealing with numerical and fact or explanatory variables - part 2
- Chapter10: Multiple linear regression
- Chapter11: Dealing with factors with more than two levels
- Chapter12: Dealing with two factors
- Chapter13: Modelling count data
- Chapter14: Modelling count data responses - two examples
- Chapter15: Modelling binary data
- Chapter16: Analysing categorical data - an introduction
- Chapter17: Analysis of contingency tables

学习提要

本门课程主要研究：线性回归模型、常见问题的解决方法

分数分布

平时分数	期末测验
20% 作业 +20% 课堂	60% 期末考试

Skip to main content

环境搭建

本课程使用工具：R Language（交互式、开放、免费）

1. 安装 R Studio
2. 安装 R Tools
3. 安装 RMarkdown 库

1. Getting Started with Regression

1.1. 什么是线性回归

线性样本回归分析：

$$\hat{y_0} = a_i + b_i x$$

原则：残差平方和最小

怎么算 a_i 和 b_i：

$$\begin{cases} b = \frac{\sum_{i=1}^{n}(x-x_i)(y-y_i)}{\sum_{i=1}^{n}(x-\bar{x})^2} \\ a = \bar{y} - b\bar{x} \end{cases}$$

1.2. 线性回归的残差与模型误差分析

残差表示预测值与真实值的差值，有正负号，一般使用 ε 表示。

$$y_i = ax_i + b + \varepsilon$$

且 ε 的值符合正态分布：$\varepsilon \sim N(0, \sigma^2)$

误差：

Skip to main content

$$
\begin{aligned}
Y - \hat{Y} &= Y - \bar{Y} - \hat{Y} + \bar{Y} \\
&= (Y - \bar{Y}) - (\hat{Y} - \bar{Y}) \\
Y - \bar{Y} &= (Y - \hat{Y}) + (\hat{Y} - \bar{Y})
\end{aligned}
$$

其中 $Y - \bar{Y}$ 称为总体差异，$Y - \hat{Y}$ 称为随机变量，$\hat{Y} - \bar{Y}$ 称为可以用自变量 x 进行解释的差异。

于是，我们有：

$$
\begin{aligned}
\sum Y - \bar{Y} &= \sum Y - \hat{Y} + \sum \hat{Y} - \bar{Y} \\
SST &= SSE + SSR \\
df = n - 1 \quad & df = n - 2 \quad df = 1
\end{aligned}
$$

并且有：

$$
\begin{cases}
MST &= \frac{SST}{df} \\
MSE &= \frac{SSE}{df} \\
MSR &= \frac{SSR}{df}
\end{cases}
$$

2. Basics of Simple Linear Regression

本课程前置需要装的包：

```
require(s20x)
```

```
Loading required package: s20x
```

2.1. 分析数据过程

2.1.1. 读取数据

读取数据表格，`header=TRUE` 表示第一行是表头，`sep=","` 表示分隔符是逗号。

Skip to main content

```r
course.df <- read.table("../data/STATS20x.txt", header = TRUE, sep = "\t")
head(course.df) # 看前面大约10行的内容
dim(course.df) # 看有多少行、多少列
course.df$Exam[1:20] # 看前20行的Exam列
```

A data.frame: 6 × 15

	Grade	Pass	Exam	Degree	Gender	Attend	Assign	Test	B	C	MC	Colc
	<chr>	<chr>	<int>	<chr>	<chr>	<chr>	<dbl>	<dbl>	<int>	<int>	<int>	<ch
1	C	Yes	42	BSc	Male	Yes	17.2	9.1	5	13	12	Bl
2	B	Yes	58	BCom	Female	Yes	17.2	13.6	12	12	17	Yell
3	A	Yes	81	Other	Female	Yes	17.2	14.5	14	17	25	Bl
4	A	Yes	86	Other	Female	Yes	19.6	19.1	15	17	27	Yell
5	D	No	35	Other	Male	No	8.0	8.2	4	1	15	Bl
6	A	Yes	72	BCom	Female	Yes	18.4	12.7	15	17	20	Bl

146 · 15

42 · 58 · 81 · 86 · 35 · 72 · 42 · 25 · 36 · 48 · 29 · 54 · 49 · 52 · 28 · 34 · 51 · 81 · 80 · 41

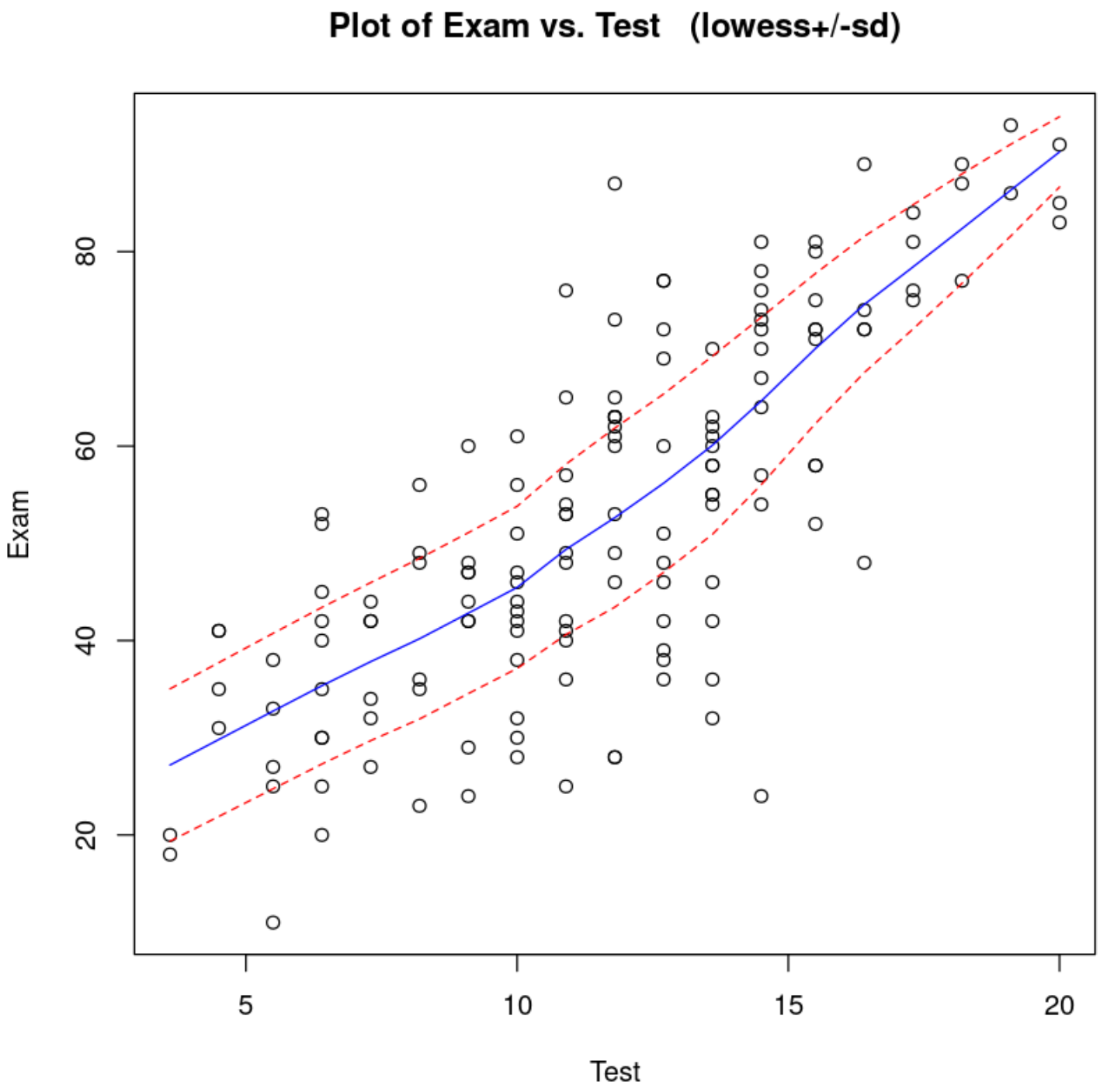
2.1.2. 绘图观测数据

对数据进行绘图分析，着重分析 `Exam` 和 `Test` 两个变量之间的关系。

首先应当粗略查看两者的关系，如线性、二次、曲线、正弦等

```r
library(s20x)
trendscatter(Exam ~ Test, data = course.df)
```

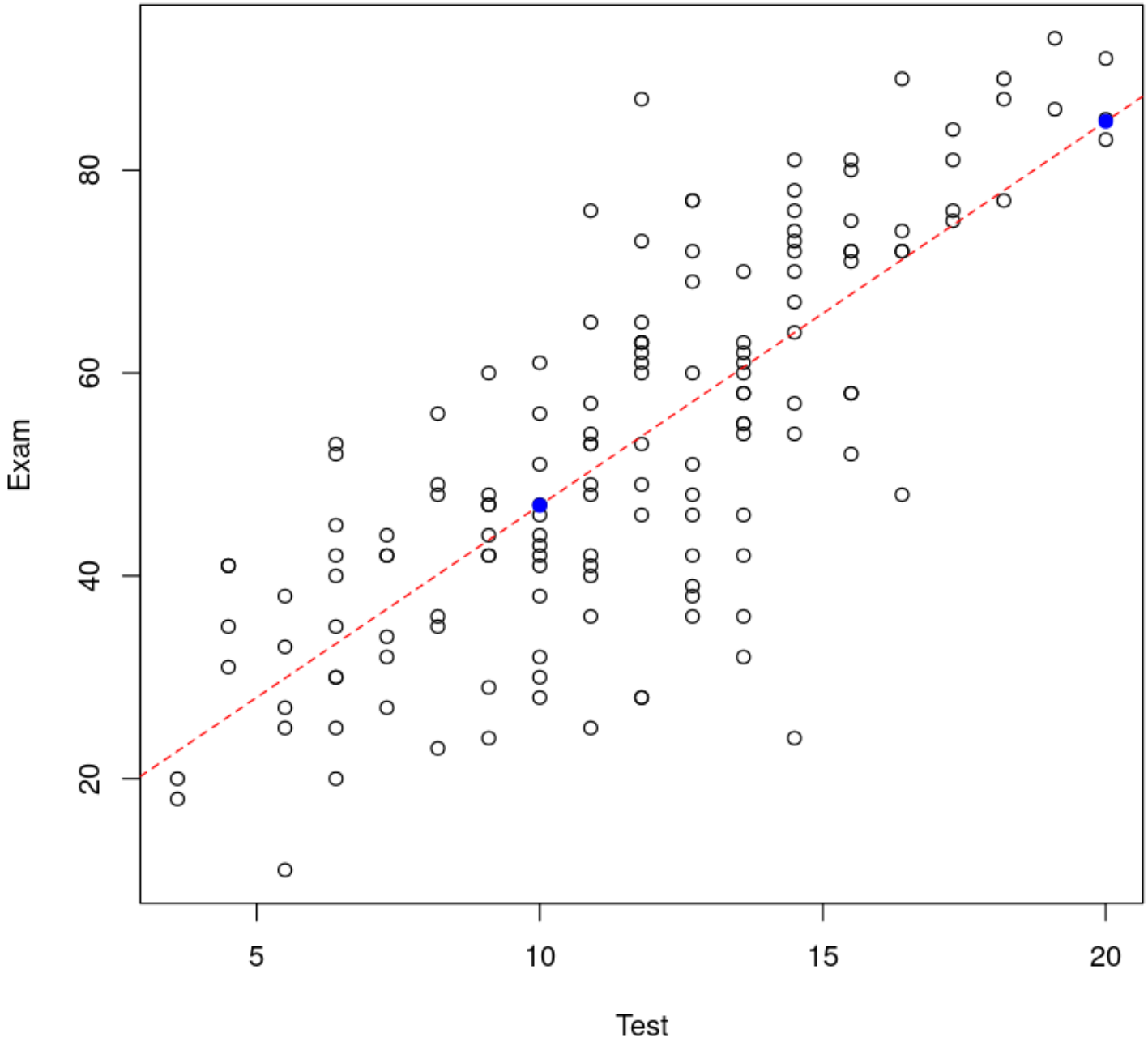
Skip to main content



2.1.3. 进行初步拟合

可以看到整体大致呈线性关系，故我们采用线性回归模型。

```r
plot(Exam ~ Test, data = course.df)
# 绘制回归直线
examtest.fit <- lm(Exam ~ Test, data = course.df)
# lty = 2 表示虚线，col = "red" 表示红色
abline(examtest.fit, lty = 2, col = "red")

points(
    0,
    predict(examtest.fit, newdata = data.frame(Test = 0)),
    col = "blue",
    pch = 19
)
points(10, predict(examtest.fit, newdata = data.frame(Test = 10)), col = "blue", pch =
points(20, predict(examtest.fit, newdata = data.frame(Test = 20)), col = "blue", pch =
```




```
summary(examtest.fit)
```

Skip to main content

```
Call:
lm(formula = Exam ~ Test, data = course.df)

Residuals:
    Min      1Q  Median      3Q     Max
-39.980  -6.471   0.826   8.575  33.242

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   9.0845     3.2204   2.821  0.00547 **
Test          3.7859     0.2647  14.301  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12.05 on 144 degrees of freedom
Multiple R-squared:  0.5868,    Adjusted R-squared:  0.5839
F-statistic: 204.5 on 1 and 144 DF,  p-value: < 2.2e-16
```

其中：

- Call：表示回归方程，指明了自变量和因变量
- Risiduals：残差，指明了残差的分布，如最大、最小、中值等
- Coefficients：系数，此处即 a_i 和 b_i 的值
- Residual standard error：残差标准差，即残差的标准差
- Multiple R-squared：多元 R^2 值
- Adjusted R-squared：调整后的 R^2 值
- F-statistic：F 统计量，即 F 统计量。F 统计量的分子是回归平方和，分母是残差平方和。F 统计量的值越大，说明回归平方和越大，即回归模型的拟合效果越好。F 统计量的值越小，说明回归平方和越小，即回归模型的拟合效果越差。p-value 则相反。

2.2. 分析数据是否可以接受

2.2.1. 残差观测

针对指定行分析预测值和残差：

```r
data.frame(course.df$Test[1], course.df$Exam[1]) # 原第一行
# 按照 tidyverse 的风格，也可以使用 dplyr 包的 select 函数来选择列
# dplyr::select(course.df[1, ], Exam, Test)
fitted(examtest.fit)[1] # 拟合值
```

Skip to main content

A data.frame: 1 × 2

course.df.Test.1.	course.df.Exam.1.
<dbl>	<int>
9.1	42

1: 43.5363712056029

1: -1.53637120560293

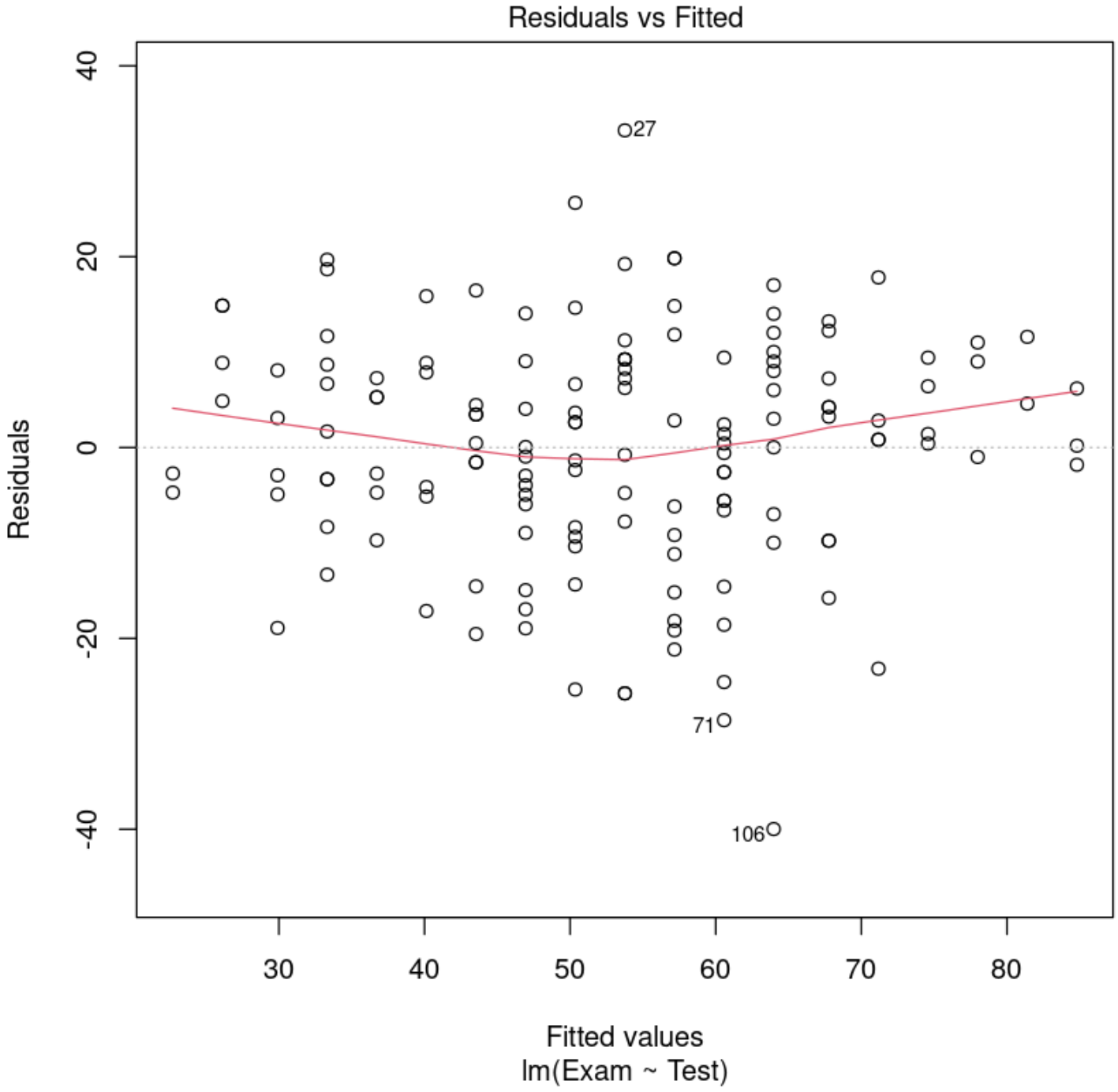
检验上，一个成功的拟合模型的残差应当有：

1. 残差均值接近于 0
2. 残差满足正态分布
3. 没有或排除了异常点

2.2.1.1. 残差均值接近于 0

分析残差，看是否符合均值等于0

```
# 其中 which = 1 表示残差直方图(histogram of residuals),
# which = 2 表示残差QQ图(qqplot,即 normal quantile-quantile-plot),
# which = 3 表示残差标准化图
plot(examtest.fit, which = 1)
```

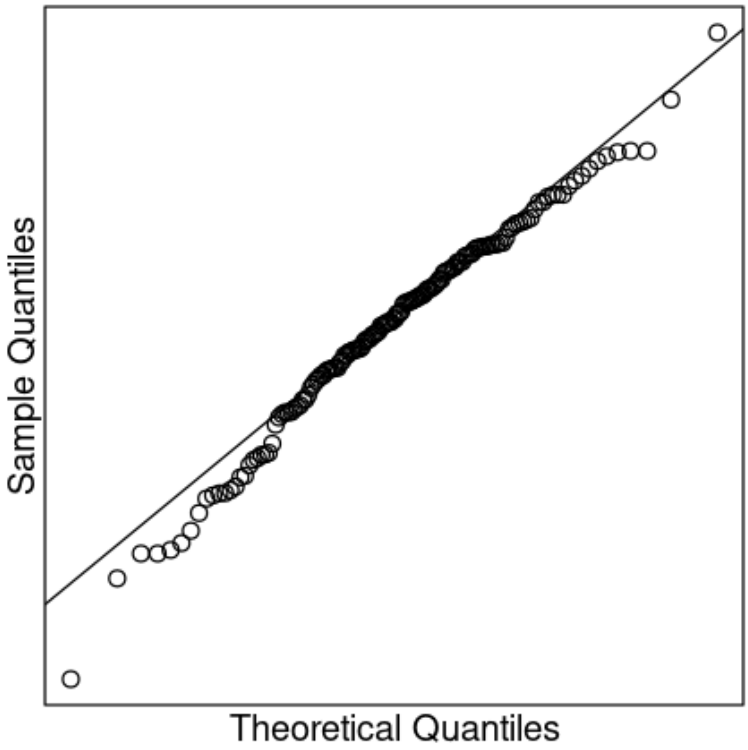


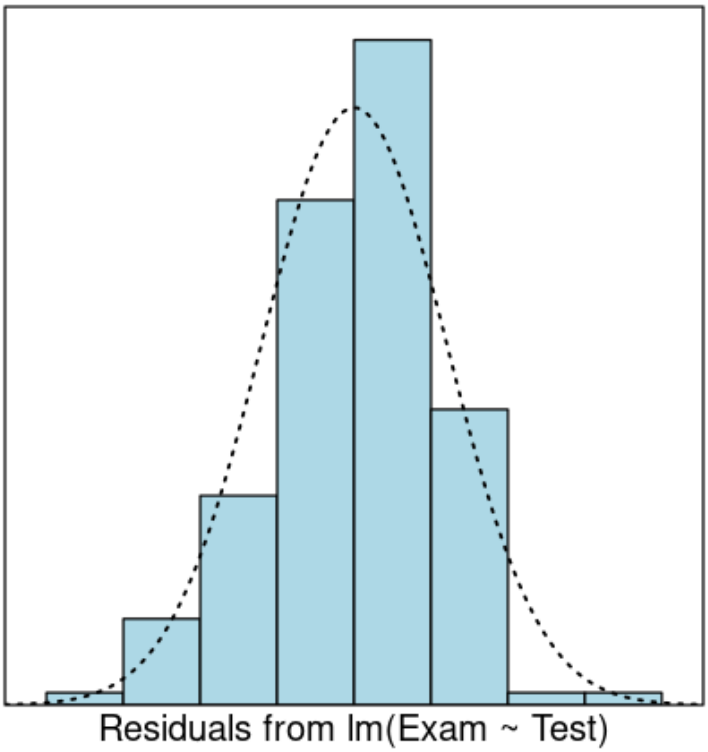
2.2.1.2. 残差满足正态分布

残差在分布上在符合正态同分布：iid – independence（并且这是根据学生在考试中应该相互独立的表现）。残差应该有大致恒定的散布。这其实是 Equality Of Variance (EOV，方差相等) 原则。

检查残差是否满足正态分布：

```
normcheck(examtest.fit)
```

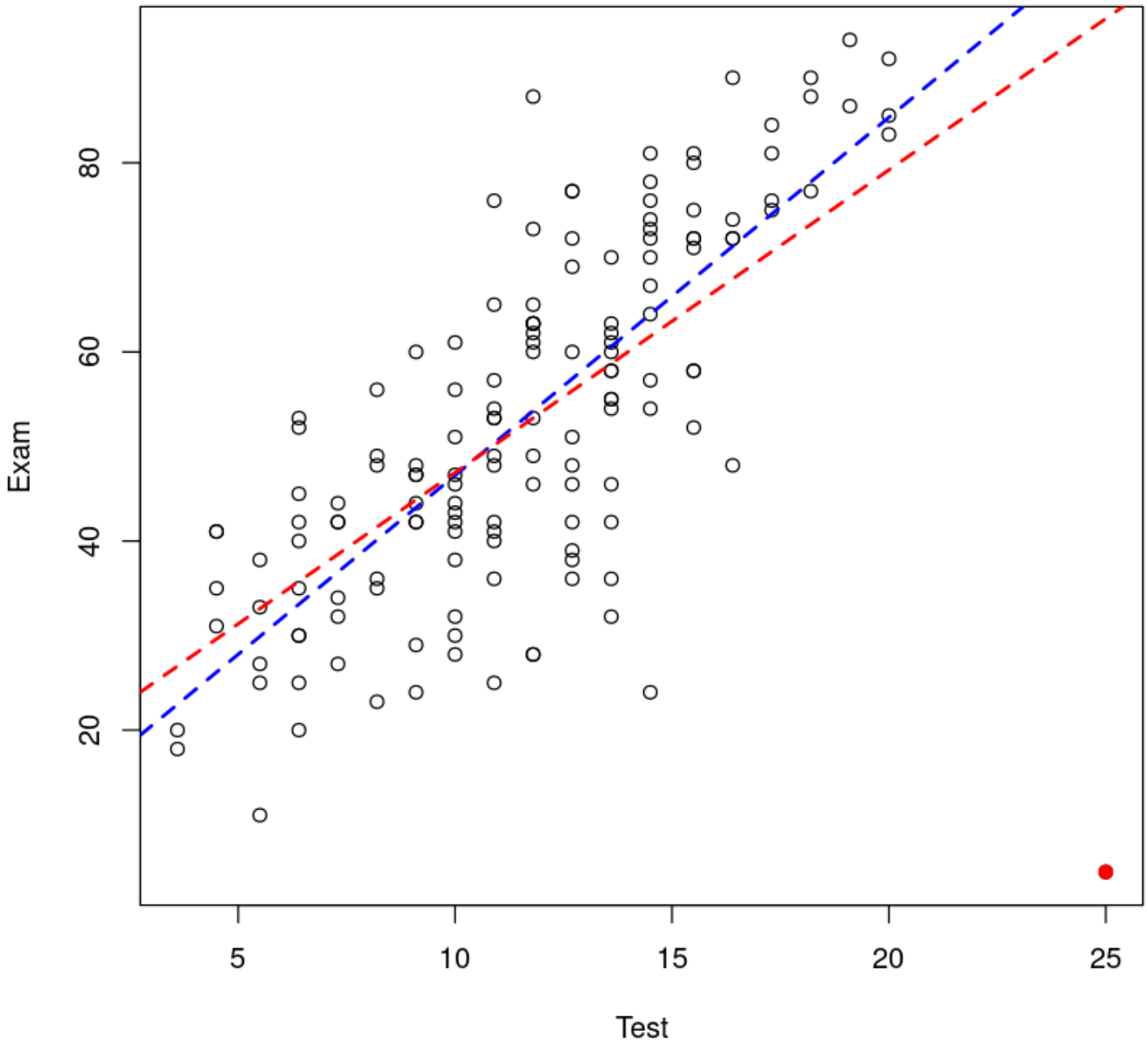




Skip to main content

```r
# 创造一个包含异常点的数据集并验证异常点对回归直线的影响
n <- nrow(course.df)
# 复制一数据集的最后一行
course2.df <- course.df[c(1:n, n), ]
# 修改新数据集的最后一行的 Test 和 Exam 列的值，故意创造一个差异极大的观测值
course2.df[n + 1, c("Test", "Exam")] <- c(25, 5)
# 画出散点图
plot(Exam ~ Test, data = course2.df)
## 并标记我们创建的新的观测点
points(25, 5, pch = 19, col = "red")

# 如果有的观测值是异常值，那么回归直线就会受到影响
examtest2.fit <- lm(Exam ~ Test, data = course2.df)
summary(examtest2.fit)

# 或者直接画图验证该点造成的影响
abline(examtest.fit, lty = 2, lwd = 2, col = "blue")
abline(examtest2.fit, lty = 2, lwd = 2, col = "red")
```

```
Call:
lm(formula = Exam ~ Test, data = course2.df)

Residuals:
    Min      1Q  Median      3Q     Max
-90.251  -6.846   2.638   9.456  33.996

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  15.2374     3.7172   4.099 6.88e-05 ***
Test          3.2006     0.3023  10.588  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14.34 on 145 degrees of freedom
Multiple R-squared:  0.436,     Adjusted R-squared:  0.4322
F-statistic: 112.1 on 1 and 145 DF,  p-value: < 2.2e-16
```

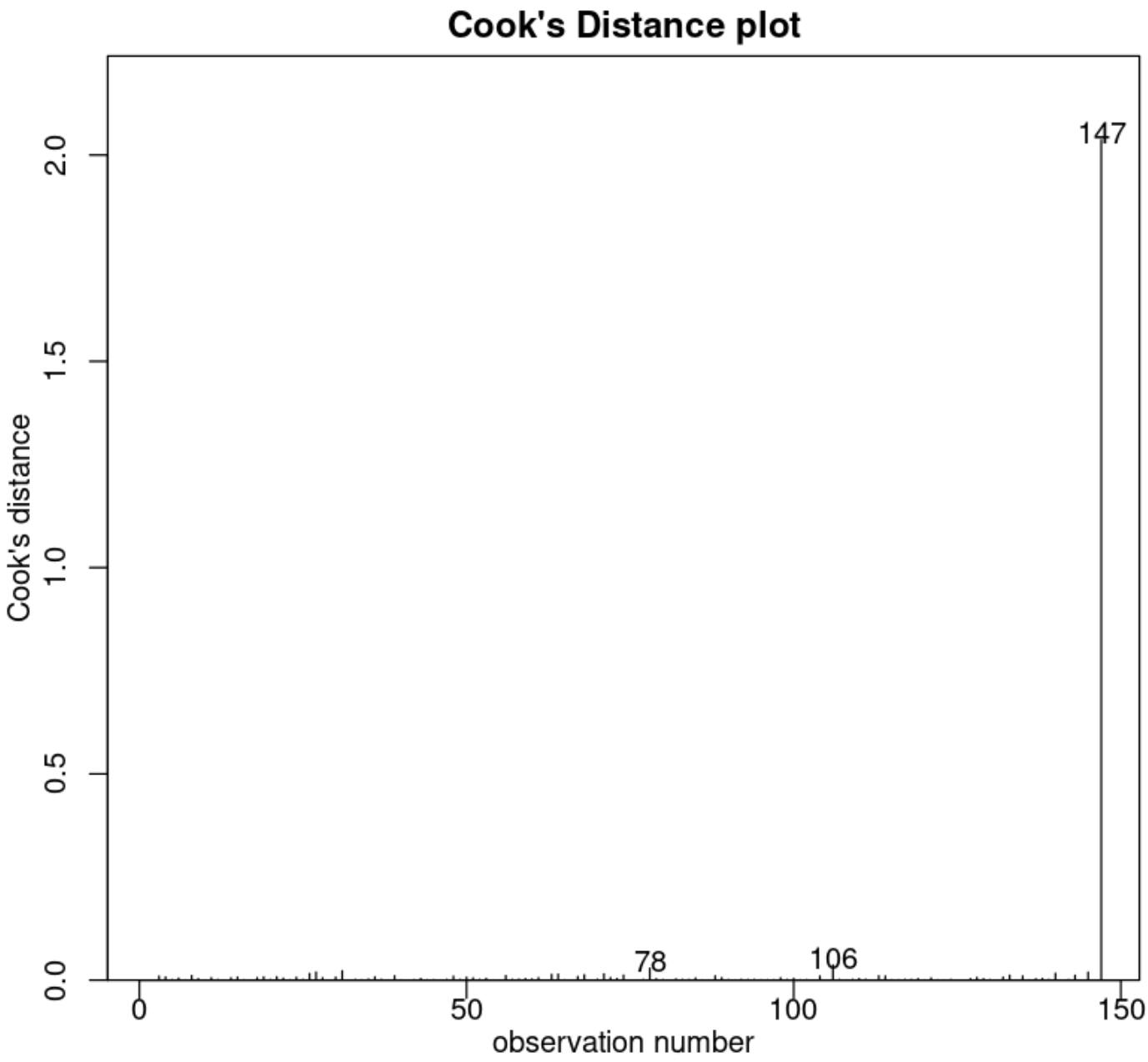


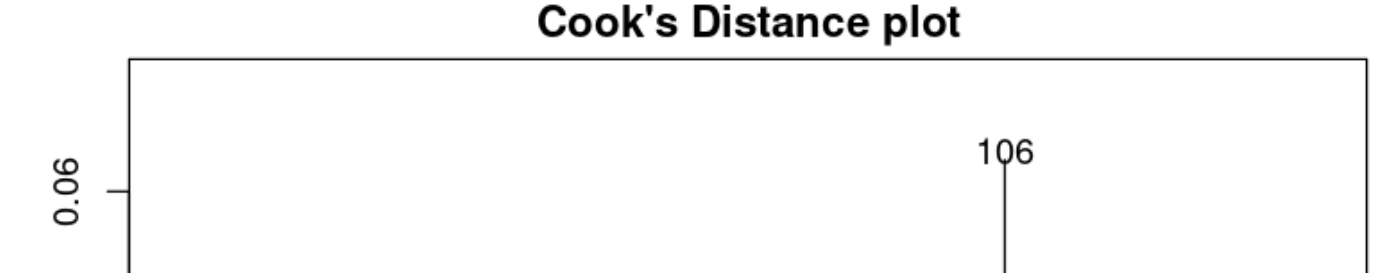
对其进行观测值差异分析：

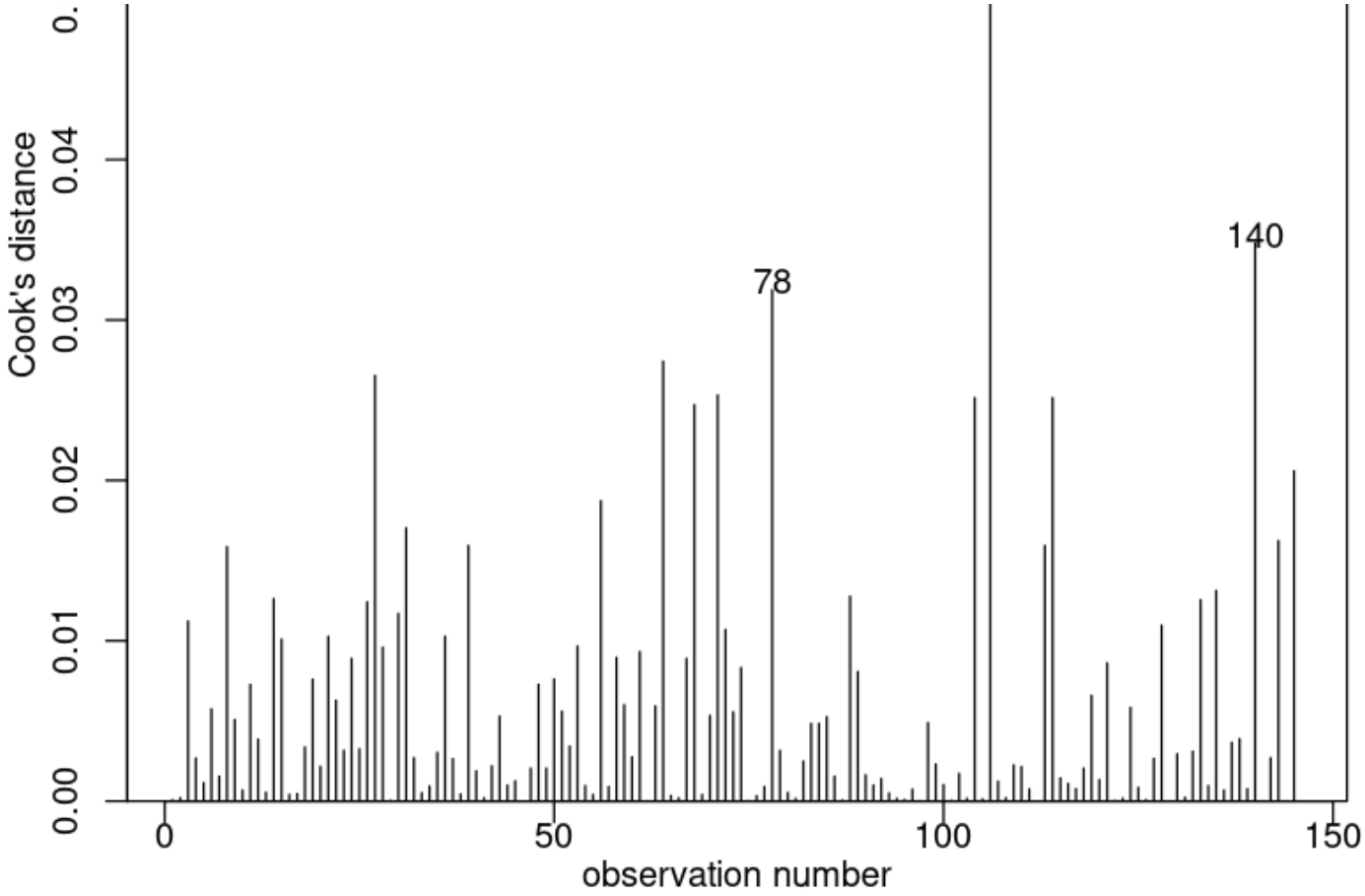
```
# 画出异常值的影响
cooks20x(examtest2.fit)
# 对比原来的值影响
cooks20x(examtest.fit)
```

Skip to main content

Skip to main content







2.2.2. R 方观测

R Squared 即 R 平方，是回归平方和与总平方和的比值，即 $R^2 = \frac{SSR}{SST}$，其中 SSR 为回归平方和，SST 为总平方和。R 平方的值越大，说明回归平方和越大，即回归模型的拟合效果越好。R 平方的值越小，说明回归平方和越小，即回归模型的拟合效果越差。

SSR 即回归平方和，是因变量的预测值与因变量的均值之差的平方和，即 $SSR = \sum_{i=1}^{n}(y_i - \bar{y})^2$，其中 y_i 为第 i 个观测值，$\bar{y}$ 为因变量的均值。下面将简要介绍 SSR 的计算方法。

```r
# 消除一次项
examnull.fit = lm(Exam ~ 1, data = course.df)
summary(examnull.fit)
# 对比之前的 Summary
summary(examtest.fit)
```

Skip to main content

```
Call:
lm(formula = Exam ~ 1, data = course.df)

Residuals:
    Min      1Q  Median      3Q     Max
-41.877 -12.877  -1.377  15.623  40.123

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   52.877      1.546   34.21   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 18.68 on 145 degrees of freedom
```

```
Call:
lm(formula = Exam ~ Test, data = course.df)

Residuals:
    Min      1Q  Median      3Q     Max
-39.980  -6.471   0.826   8.575  33.242

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   9.0845     3.2204   2.821  0.00547 **
Test          3.7859     0.2647  14.301  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12.05 on 144 degrees of freedom
Multiple R-squared:  0.5868,    Adjusted R-squared:  0.5839
F-statistic: 204.5 on 1 and 144 DF,  p-value: < 2.2e-16
```

此时我们可以得到 SS (Null) 的值 18.68，以及 SS (Test) 的值 12.05。

R 方的值即 1 - SS (Null) /SS (Test) 的值，即 0.5868。

置信区间：$[a_i - 2SE(a_i), a_i + 2SE(a_i)]$，即 $[a_i - 2\sqrt{Var(a_i)}, a_i + 2\sqrt{Var(a_i)}]$，其中 $Var(a_i)$ 为 a_i 的方差。

2.2.3. 每一个拟合值的 T 检验

> 知道看什么，什么意思，怎么看

```r
summary(examtest.fit)
```

```
Call:
lm(formula = Exam ~ Test, data = course.df)

Residuals:
    Min      1Q  Median      3Q     Max
-39.980  -6.471   0.826   8.575  33.242

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   9.0845     3.2204   2.821  0.00547 **
Test          3.7859     0.2647  14.301  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12.05 on 144 degrees of freedom
Multiple R-squared:  0.5868,    Adjusted R-squared:  0.5839
F-statistic: 204.5 on 1 and 144 DF,  p-value: < 2.2e-16
```

可以看出 Test 行的 Pr（P-value）的值小于 2.2x10^-16，远小于 0.05，故拒绝原假设，即拟合值的系（旁边的3颗*也表示可信度极高，即该斜率的线性拟合极好）

- 零假设 H_0：Test 和 Exam 之间的线性关系系数为 0（没有线性关系），即 即 a_i 的系数为 0
- 备择假设 H_1：Test 和 Exam 之间的线性关系系数不为 0（有线性关系），即 即 a_i 的系数不为 0

我们对于斜率的置信程度，是由标准误差决定的，即 $SE(a_i)$，即 $SE(a_i) = \sqrt{\frac{SSE}{n-2}}$，其中 SSE 为残差平方和，即 $SSE = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$，其中 $\hat{y}_i$ 为第 i 个观测值的预测值，即 $\hat{y}_i = a_i + b_i x_i$，$x_i$ 为第 i 个观测值的自变量值。此处的 $se(a)$ 为 0.2647。于是我们有：

$$\frac{3.7859 - 0}{0.2647} = 14.34$$

此结果表示偏离此结果的标准差，这个数字越大，代表我们对于斜率的置信程度越高。

2.3. 利用分析结果做预测

2.3.1. 拟合值的置信区间

Skip to main content

```r
confint(examtest.fit)
# Intercept 即截距，Test 即斜率
# 也可以自己修改置信水平
confint(examtest.fit, level = 0.99)
```

A matrix: 2 × 2 of type dbl

	2.5 %	97.5 %
(Intercept)	2.719020	15.449907
Test	3.262659	4.309189

A matrix: 2 × 2 of type dbl

	0.5 %	99.5 %
(Intercept)	0.6778171	17.491110
Test	3.0948635	4.476984

2.3.2. 预测

1. 准确预测值
2. 预测的均值范围
3. 预测每一个个体的取值范围

区间估计和点估计的区别：

- 区间估计：给出一个区间，表示参数的可能取值范围
- 点估计：给出一个点，表示参数的可能取值

```r
# 区间估计
preds.df <- data.frame(Test = seq(0, 20, by = 10))
predict(examtest.fit, newdata = preds.df, interval = "confidence")
# 点估计
predict(examtest.fit, newdata = preds.df, interval = "prediction")
```

A matrix: 3 × 3 of type dbl

	fit	lwr	upr
1	9.084463	2.71902	15.44991
2	46.943703	44.80912	49.07828
3	84.802942	79.97021	89.63568

A matrix: 3 × 3 of type dbl

	fit	lwr	upr
1	9.084463	-15.56475	33.73368
2	46.943703	23.03510	70.85231
3	84.802942	60.50438	109.10151

其中：

- 区间估计表格的 [2,2:3] 表示所有半期考试10分，期末考试的分数的均值的范围
- 区间估计表格的 [2,2:3] 表示所有半期考试10分个体的分数的范围，落在这个范围即为正常值

2.4. 总结

遇到此类问题，通用思路（适用于分析x和y两个未知数的某种关系）：

- 绘制数据散点图并简要查看自变量与因变量之间是哪种关系（如果有关系），最好是能够通过工具分析（也可能会有一份研究意图的声明可以被指导）。提出适当的研究方式。在上边的例子中，我们就决定采用了线性模型：

$$y = \beta_0 + \beta_1 x_i + \varepsilon_i, \varepsilon_i \sim N(0, \sigma^2)(where \beta_1 > 0)$$

- 使用 `lm` 函数进行模型拟合。
- 检查我们提出的假设进行合适方式的验证。
  - Independence OK? (how were the data collected?)
  - EOV Okay? Using `plot(examtest.fit, which = 1)`.
  - Normality Okay? Using `normcheck`.

If these are okay then go to next step

- 尝试适时删除任何不重要的解释变量（后面会讲）。如果能删除，请检查新的研究方式。
- 确保个别要点不会产生过分的不适当的影响，并尝试删除/纠正它们。Using `cooks20x`.
- 做出结论/预测，讨论极限，并回答相关的研究问题。

注意：在上述步骤中，在对当前步骤满意之前，切记不要匆忙进行下一步。

3. The null model

本课程前置需要装的包：

```
require(s20x)
require(bootstrap)
```

```
Loading required package: s20x
```

```
Loading required package: bootstrap
```

```
Warning message in library(package, lib.loc = lib.loc, character.only = TRUE, logical.re
“there is no package called ‘bootstrap’”
```

3.1. Revisiting the null model 回顾零模型

本节同样以 Stats20x 的学生考试成绩为例：

```
Stats20x.df <- read.table("../data/STATS20x.txt", header = TRUE, sep = "\t")
```

零模型就是把线性模型中的斜率去掉，或斜率指定常数，从而排除其影响单独分析截距。本节将重点讲述零模型的最大作用：T检验。

> 一文详解t检验 - 知乎
>
> t检验（t test）又称学生t检验（Student t-test）可以说是统计推断中非常常见的一种检验方法，用于统

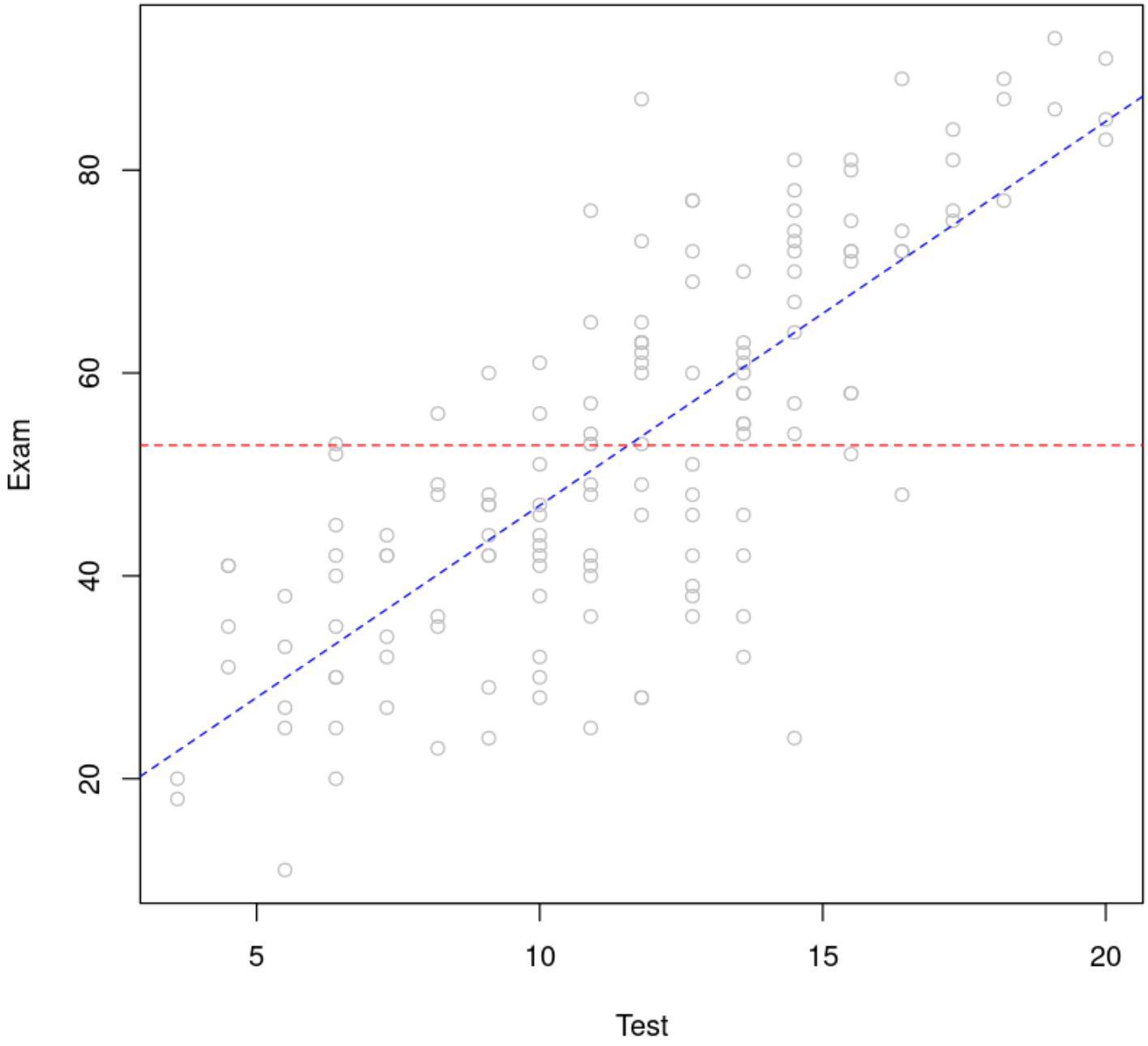
Skip to main content

t检验的前提是要求样本服从正态分布或近似正态分布，不然可以利用一些变换（取对数、开根号、倒数等等）试图将其转化为服从正态分布是数据，如若还是不满足正态分布，只能利用非参数检验方法。不过当样本量大于30的时候，可以认为数据近似正态分布。

t检验最常见的四个用途：

- 单样本均值检验（One-sample t-test）用于检验 “总体方差未知、正态数据或近似正态的” 单样本的均值，是否与已知的总体均值相等。
- 两独立样本均值检验（Independent two-sample t-test）用于检验两对 “独立的，正态数据或近似正态的” 样本的均值是否相等，这里可根据总体方差是否相等分类讨论。
- 配对样本均值检验（Dependent t-test for paired samples）用于检验一对配对样本的均值的差，是否等于某一个值
- 回归系数的显著性检验（t-test for regression coefficient significance）用于检验回归模型的解释变量，对被解释变量是否有显著影响

```
# 建立回归模型
examtest.fit <- lm(Exam ~ Test, data = Stats20x.df)
examtest.fit2 <- lm(Exam ~ 1, data = Stats20x.df)

# 绘图
plot(Exam ~ Test, data = Stats20x.df, col = "grey")
abline(examtest.fit, col = "blue", lty = 2)
abline(examtest.fit2, col = "red", lty = 2)
```

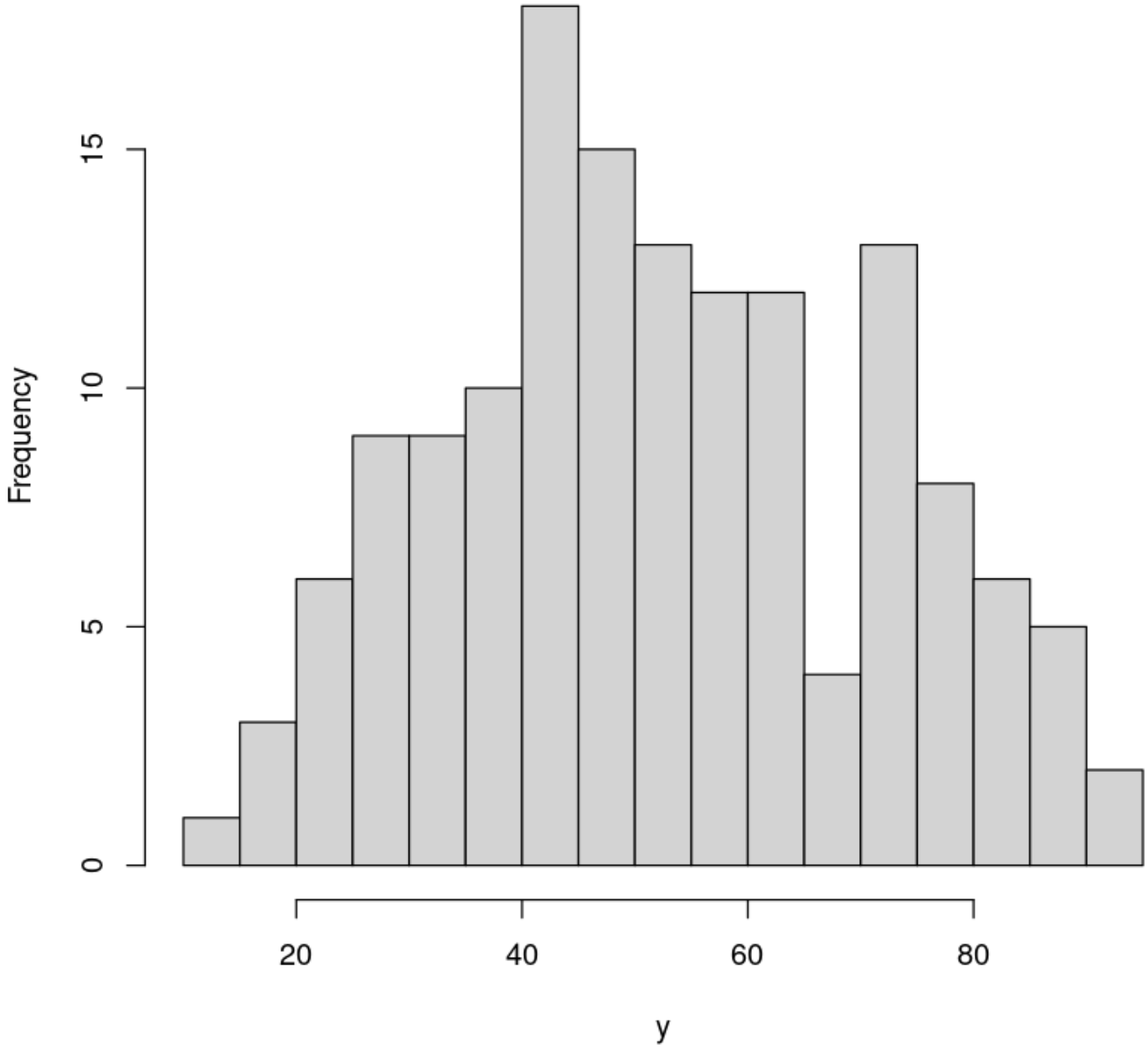


推断总体均值：

To save some typing we'll let y be the vector Stats20x.df$Exam of exam scores.

```
y=Stats20x.df$Exam
hist(y,breaks=20,main="") #Use main to suppress plot title
```

Skip to main content



继续使用零模型做线性回归，使其更关注于y值的置信关系与p检验。

```
null.fit = lm(y ~ 1)
# Only give coefficients from summary 将系数板块单独提取出做展示
coef(summary(null.fit))
# 获得该零模型的对应置信区间
confint(null.fit)
```

A matrix: 1 × 4 of type dbl

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	52.87671	1.545802	34.20666	2.632011e-71

A matrix: 1 × 2 of type dbl

	2.5 %	97.5 %
(Intercept)	49.8215	55.93193

Conclusion:

- The near zero $Pr(>|t|)$ p-value totally rejects(拒绝) the null hypothesis(零假设) that $H0 : \mu \equiv \beta 0 = 0$.
- The 95% confidence interval(置信区间) for μ is 49.82 to 55.93.

3.2. Revisiting the t-test

$$T = \frac{\bar{y} - \mu}{\frac{s}{\sqrt{n}}} \sim t_{n-1}$$

其中 $\bar{y}$ 为样本均值，s 为样本标准差。

$$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2}$$

```
n=length(y) #146 students
tstat=(mean(y)-0)/(sd(y)/sqrt(n))
tstat
```

34.2066579217089

Skip to main content

```
## t-multiplier
tmult = qt(1-.05/2, df=n-1)
## We want the upper 97.5% (or 1-.05/2) bound of the CI
## NOTE: mean = sample mean; sd = standard deviation; sqrt = square root
mean(y) - tmult * sd(y) / sqrt(n)

## Upper bound of CI 置信区间上限
mean(y) + tmult*sd(y)/sqrt(n)
## Or if we want both the lower and upper bounds of the CI in one statement
## 置信区间下限
mean(y) + c(-1,1)*tmult*sd(y)/sqrt(n)
```

49.8214976403875

55.9319270171467

49.8214976403875 · 55.9319270171467

零模型就是单样本T检验。

手动随机抽样检验我们的结果：

```
## Resampling the exam marks, N times with replacement:
N <- 10000 # The number of bootstrap resamples we want
# The new sample means are stored in ybar
ybar <- rep(NA, N) ## A vector of length N to store our resampled means

## A loop - allows us to do something N (10,000) times
for (i in 1:N) {
    ## Take the average of this sample (below) from a sample of size n = 146 from y - w
    ybar[i] <- mean(sample(y, n, replace = T))
}
mean(ybar)
```

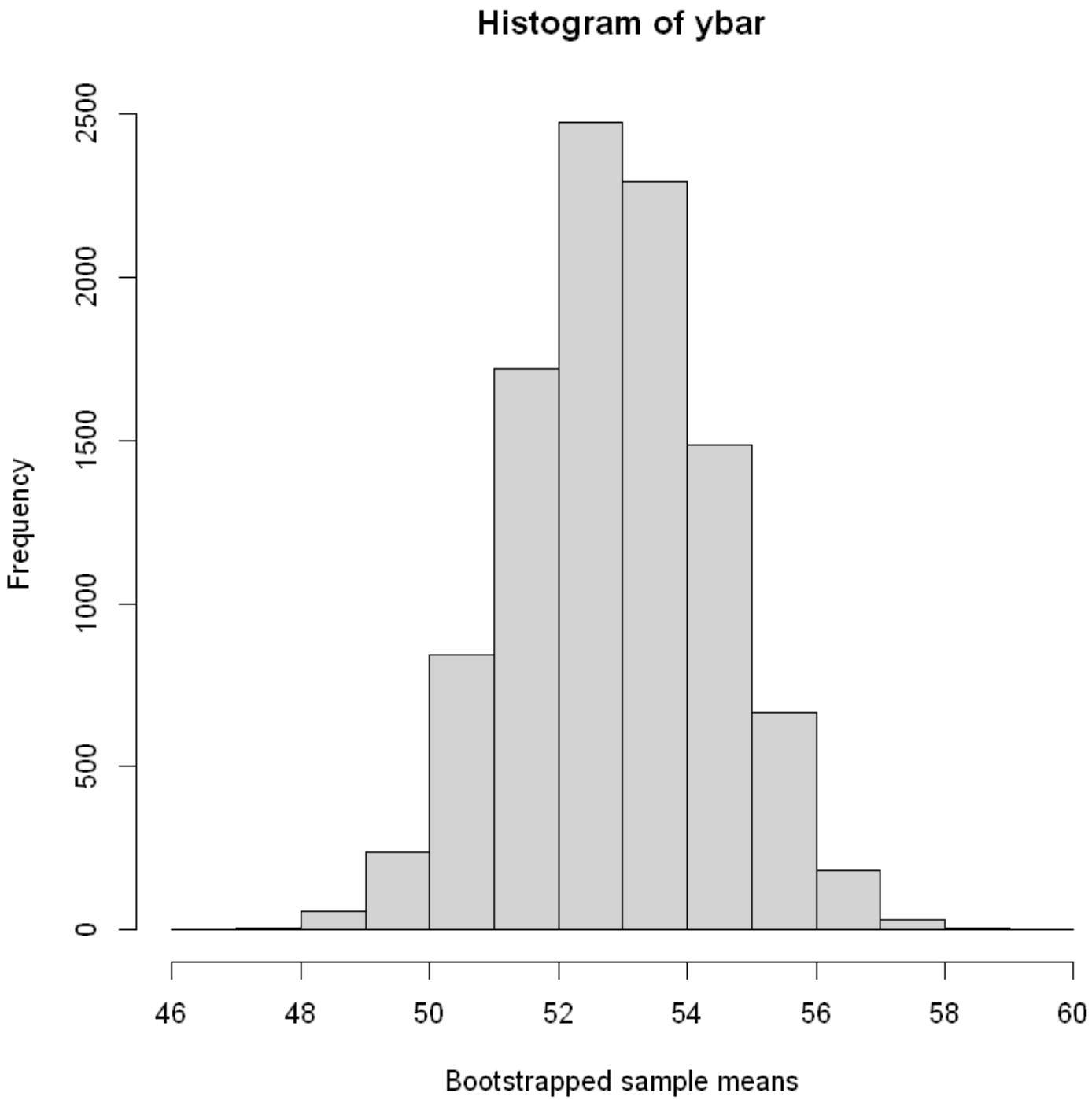
52.8913198630137

```
library(bootstrap)
ybar = bootstrap(Stats20x.df$Exam, 10000, mean)$thetastar
mean(ybar)
```

```
Error in library(bootstrap): there is no package called ‘bootstrap’
Traceback:

1. library(bootstrap)
```

Skip to main content

```r
## Histogram of these 10,000 bootstrap means
hist(ybar,xlab="Bootstrapped sample means")
```



3.3. The paired t-test

Skip to main content

For a meaningful comparison, We will need to make them have the same scale, so we multiply the test mark by 5 so that it is also out of 100.

```
Stats20x.df$Test2 = 5 * Stats20x.df$Test
## Check that it worked
Stats20x.df[1:3, c("Exam","Test","Test2")]
```

A data.frame: 3 × 3

	Exam	Test	Test2
	<int>	<dbl>	<dbl>
1	42	9.1	45.5
2	58	13.6	68.0
3	81	14.5	72.5

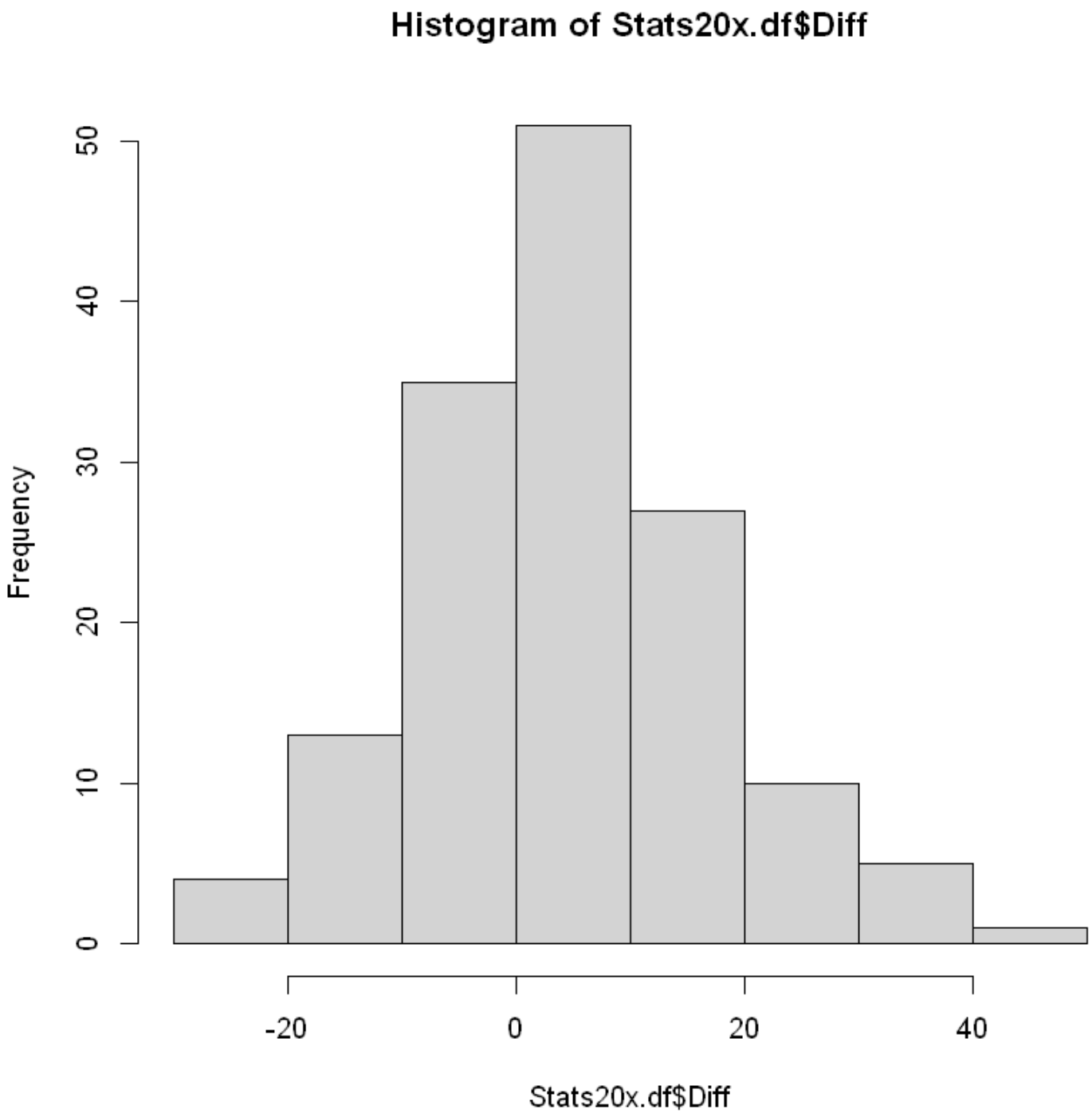
```
Stats20x.df$Diff = Stats20x.df$Test2 - Stats20x.df$Exam
## Check the first 5 measurements
Stats20x.df[1:5, c("Test2","Exam","Diff")]
```

A data.frame: 5 × 3

	Test2	Exam	Diff
	<dbl>	<int>	<dbl>
1	45.5	42	3.5
2	68.0	58	10.0
3	72.5	81	-8.5
4	95.5	86	9.5
5	41.0	35	6.0

```
hist(Stats20x.df$Diff)
```

Skip to main content



4. Fitting curves with the linear model

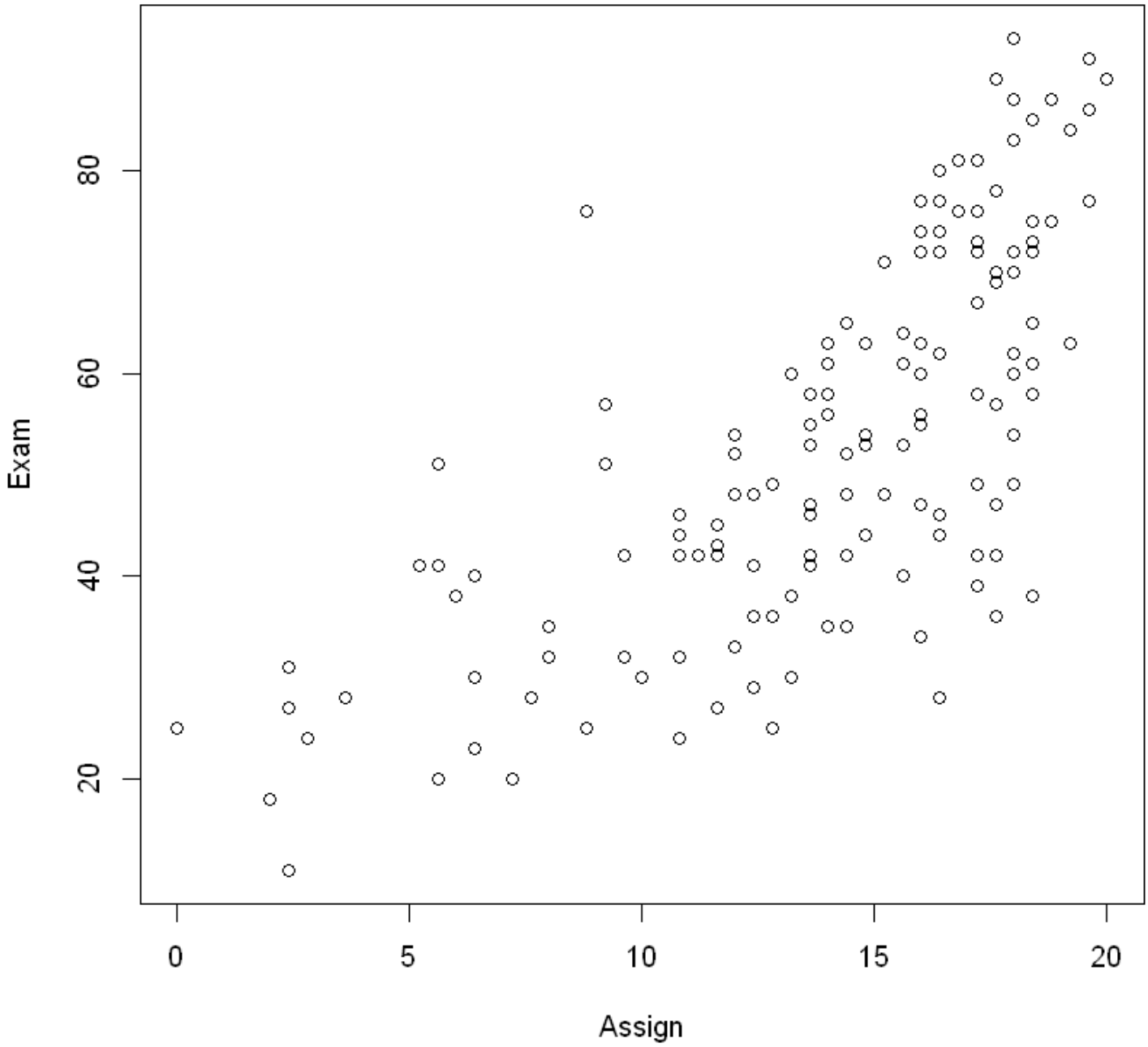
本节需要的包：

```
require(s20x)
```

```
载入需要的程辑包：s20x
```

4.1. Identifying a curved relationship 初步探究曲线关系

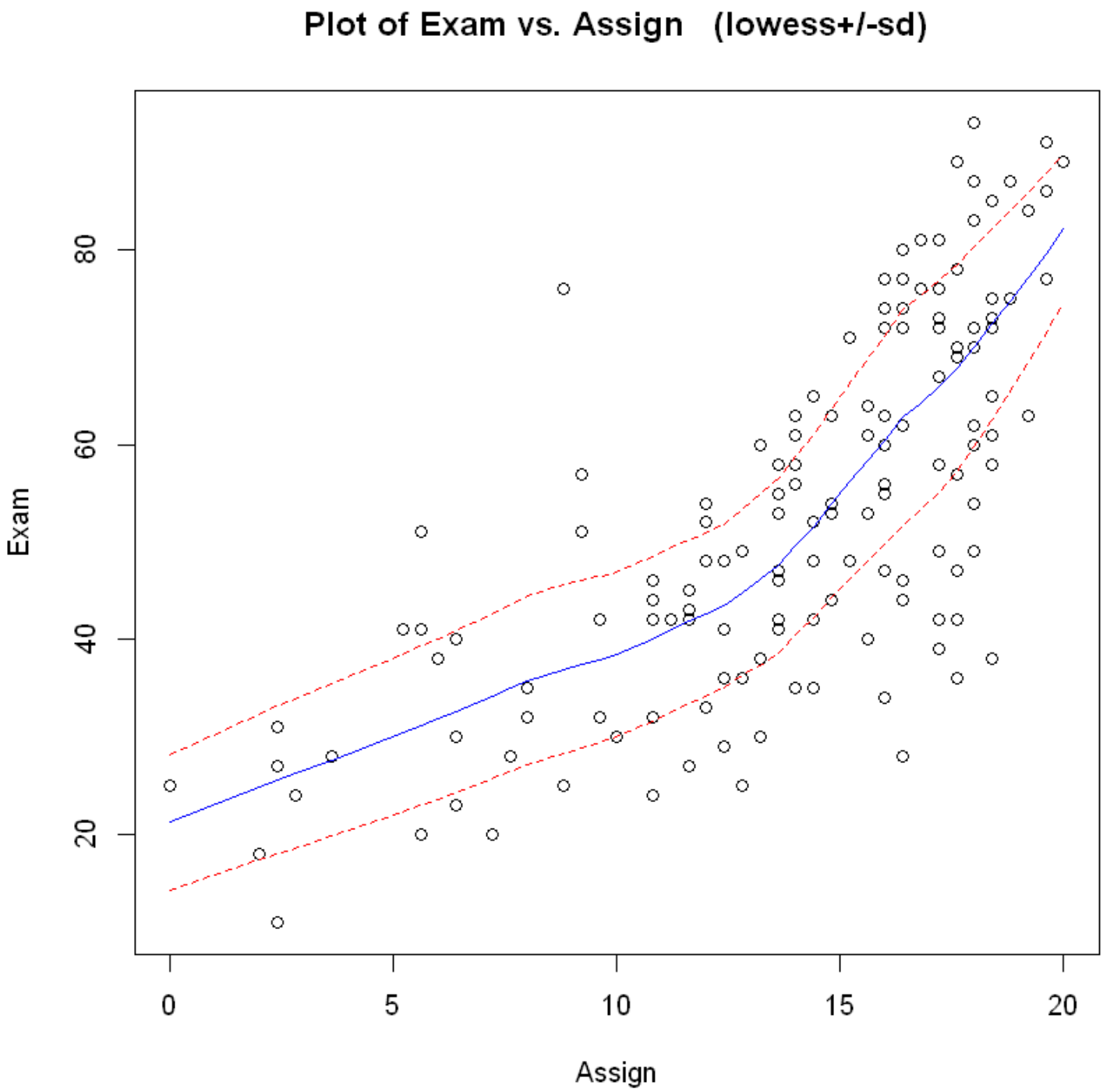
```
## Load the s20x library into our R session
library(s20x)
## Importing data into R
Stats20x.df = read.table("../data/STATS20x.txt", header=T)
## Examine the data
plot(Exam ~ Assign, data = Stats20x.df)
```



Hmmm, not quite a straight line – could be some curvature. Maybe will paint a clearer picture. 不是一条很直的线–可能是一些曲率。也许会描绘出一幅更清晰的图景。

```
trendscatter(Exam ~ Assign, data = Stats20x.df)
```

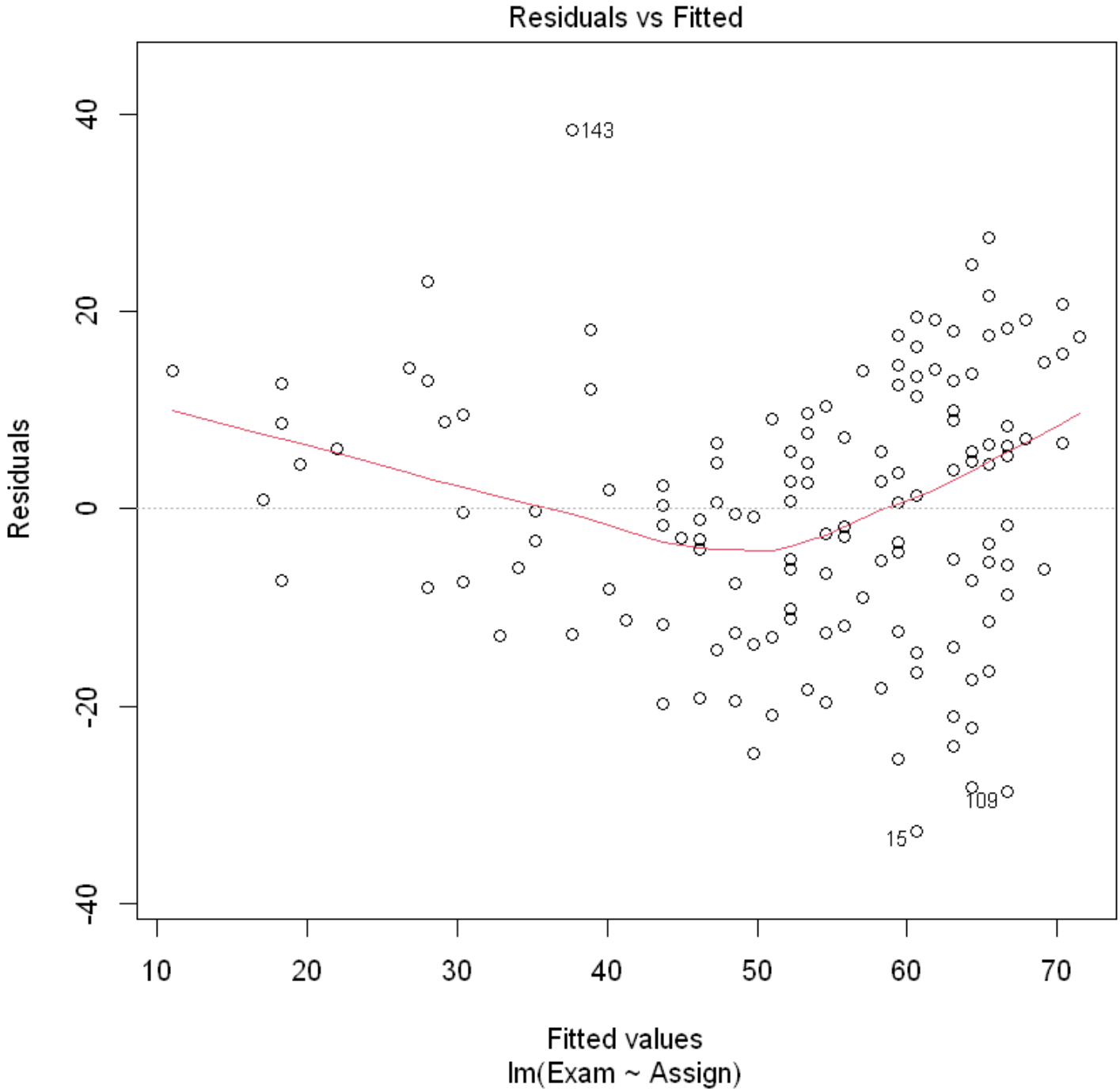
Skip to main content



Let's fit a simple linear model to these data and see if it works out or not.

```
examassign.fit = lm(Exam ~ Assign, data = Stats20x.df)
plot(examassign.fit, which = 1)
```

Skip to main content



The assumption of identical distribution with expected value of 0 looks to be questionable here. There tend to be more negative residuals in the middle, but more positive residuals at the extremes of the fitted values. Potential solution – add a quadratic (squared term) for.

假设相同的分布与预期值0看起来可疑的。会有更多负面的残差在中间,但更积极的残差的极端值。潜在的解决方案应该是：添加一个二次项(平方项)。

Skip to main content

4.2. Fitting a quadratic model 拟合二次模型

The standard notation for a quadratic curve is:

$$y = ax^2 + bx + c$$

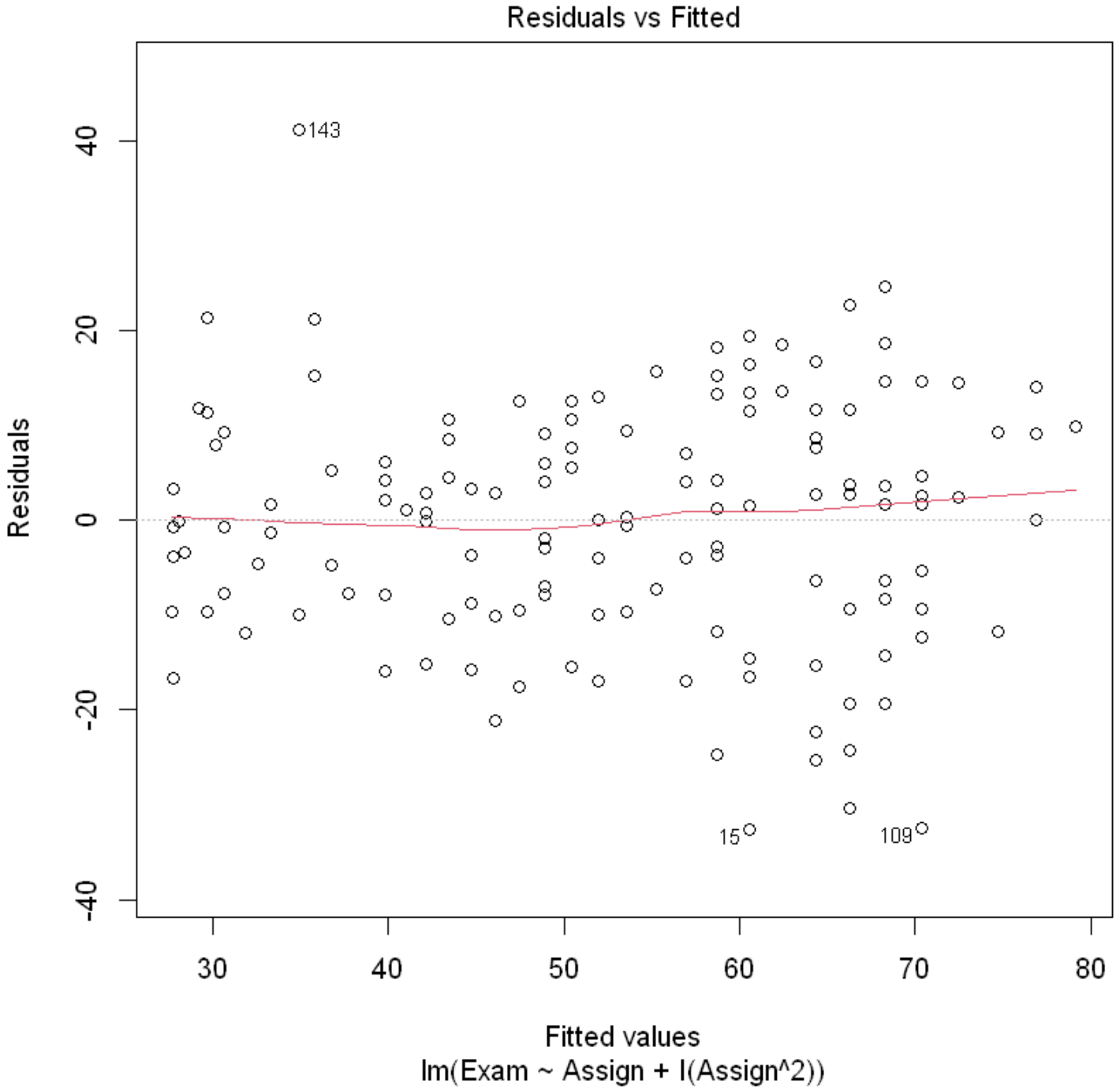
Here we will use different notation: $\beta_0 = c$, $\beta_1 = b$ and $\beta_2 = a$ and use the quadratic curve to describe the expected value of our dependent variable y. That is, we will use the following notation:

$$E[Y|x] = \beta_0 + \beta_1 x + \beta_2 x^2$$

> If $\beta_2 > 0$, then the quadratic has slope that increases with increasing x(斜率随着x增大而增大). If $\beta_2 < 0$, then the quadratic has slope that decreases with increasing x. If $\beta_2 = 0$, then the quadratic(该“二次曲线”) has a constant slope(倾斜直线的外观).

让我们回到之前的学生数据集。我们将使用一个新的变量 x^2 来拟合一个二次模型：

```
examassign.fit2 = lm(Exam ~ Assign + I(Assign^2), data = Stats20x.df)
plot(examassign.fit2, which = 1)
```



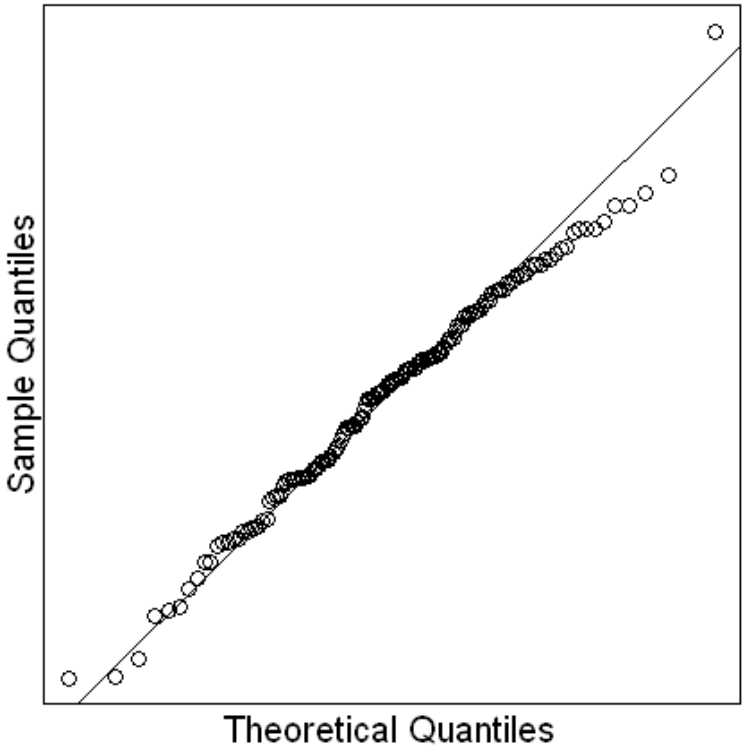
That is looking much better.

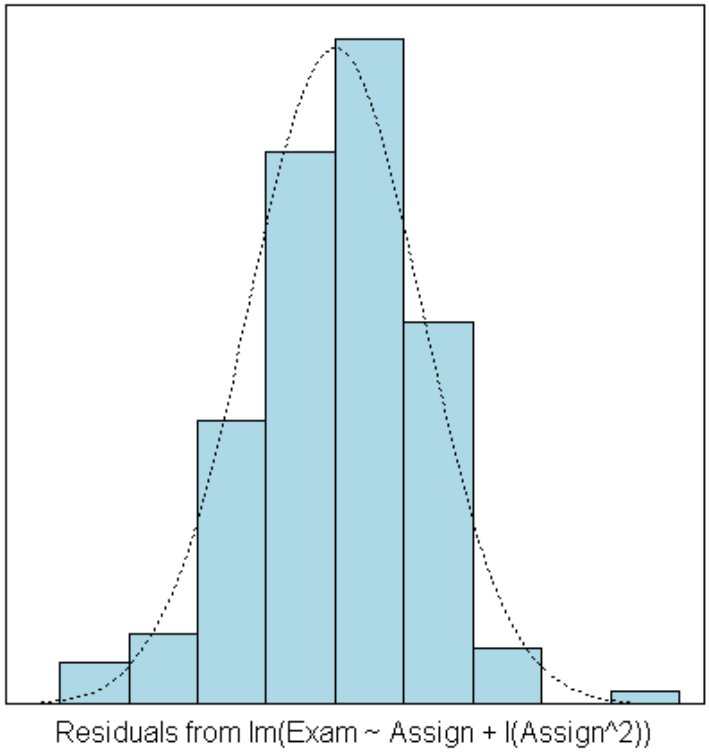
接下来我们会进行“三步走”中的后两步：

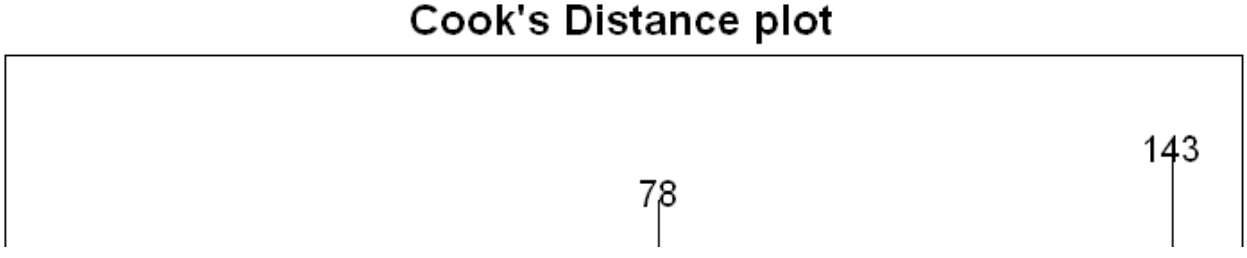
```
normcheck(examassign.fit2)
cooks20x(examassign.fit2)
```

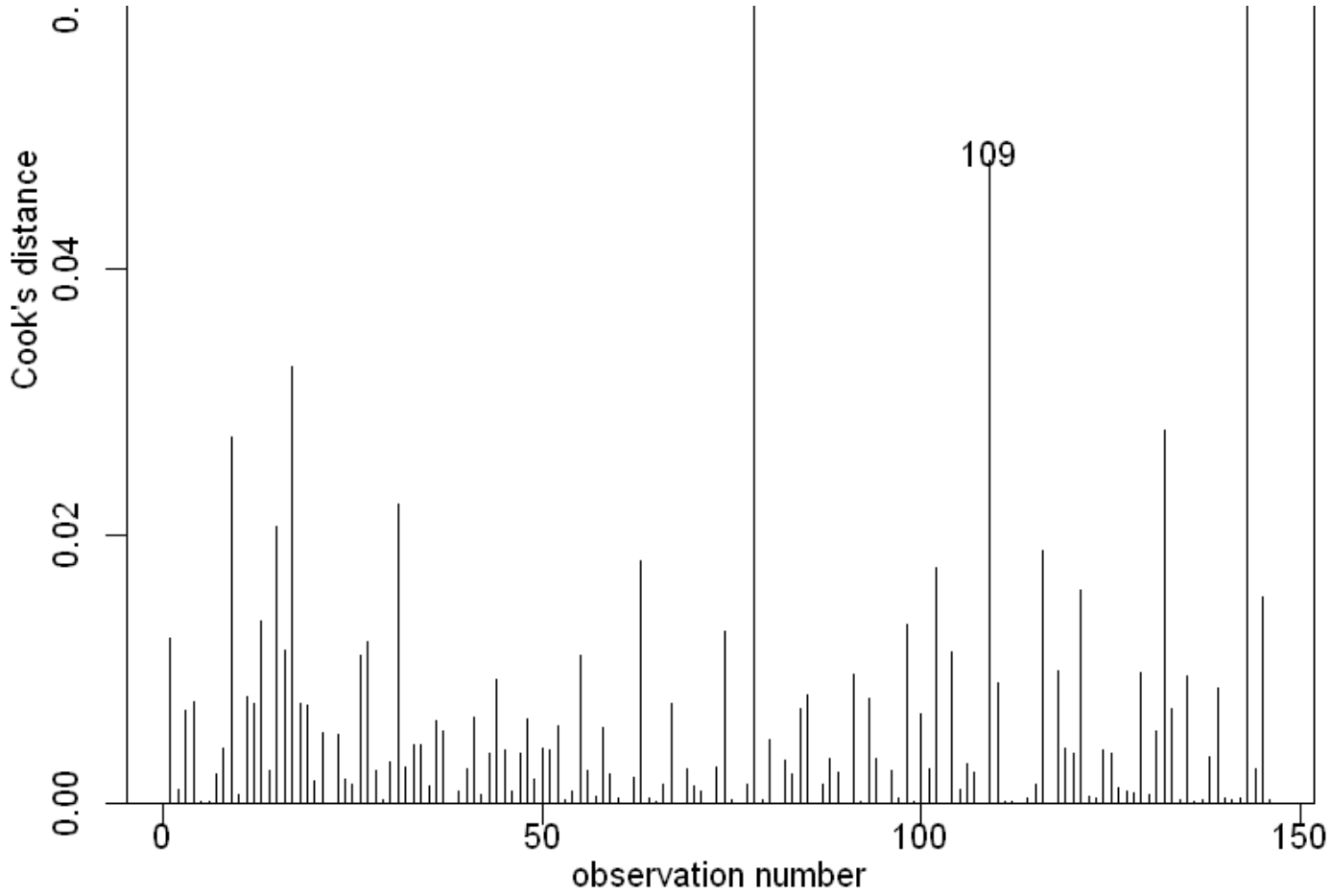
Skip to main content

Skip to main content



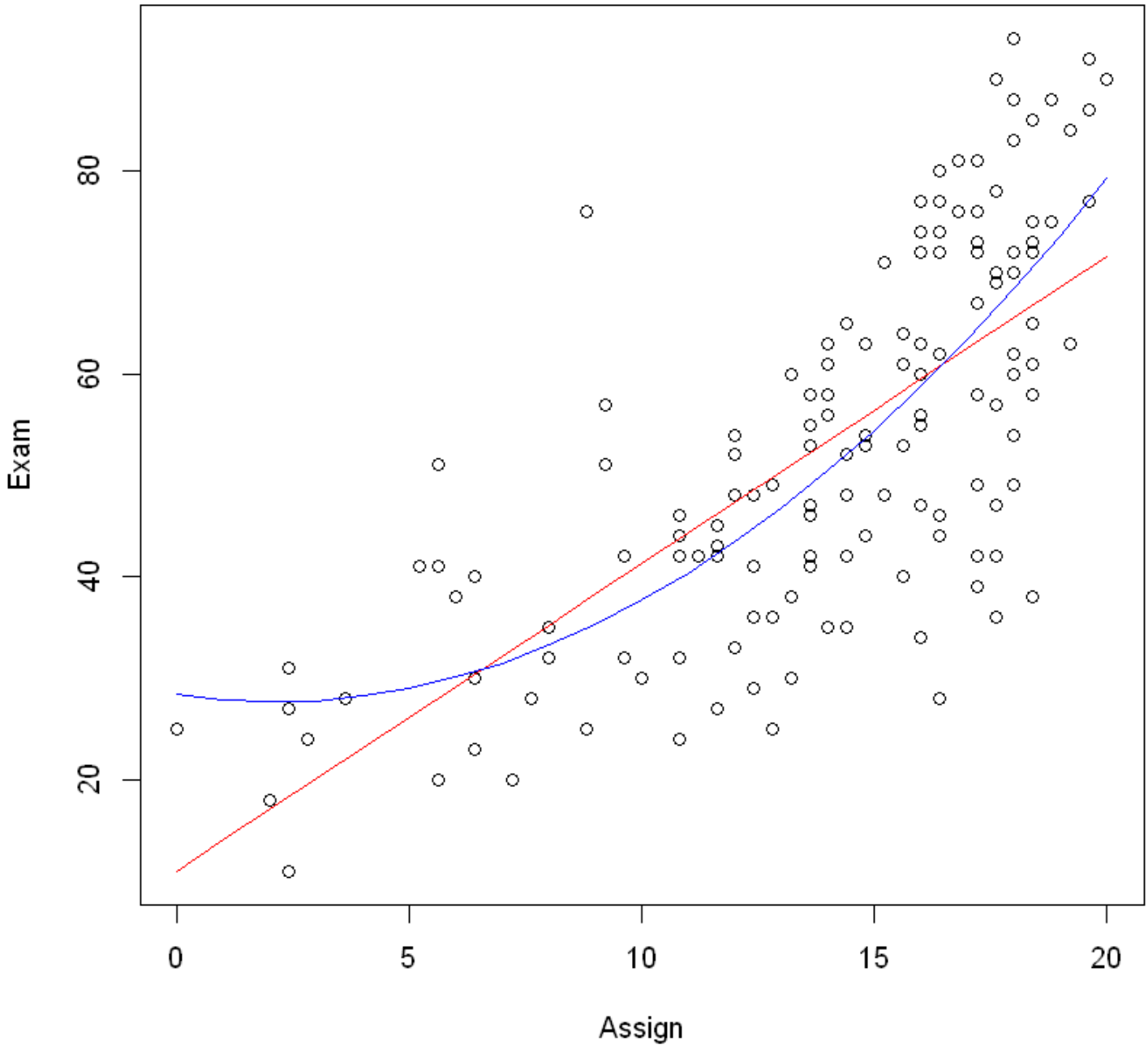






符合正态分布、方差齐性。我们可以尝试对照一下原来的模型和我们的新模型：

```
plot(Exam ~ Assign, data = Stats20x.df)
x=0:20 #Assignment values at which to predict exam mark
## Plot model 1
lines(x, predict(examassign.fit,data.frame(Assign=x)), col="red")
## Plot model 2
lines(x, predict(examassign.fit2,data.frame(Assign=x)), col="blue")
```




```
summary(examassign.fit2)
```

Skip to main content

```
Call:
lm(formula = Exam ~ Assign + I(Assign^2), data = Stats20x.df)

Residuals:
    Min      1Q  Median      3Q     Max
-32.541  -9.149   1.273   9.087  41.116

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 28.41396    5.99081   4.743 5.05e-06 ***
Assign      -0.68172    1.07242  -0.636 0.525999
I(Assign^2)  0.16102    0.04545   3.542 0.000536 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.65 on 143 degrees of freedom
Multiple R-squared:  0.5477,    Adjusted R-squared:  0.5414
F-statistic: 86.59 on 2 and 143 DF,  p-value: < 2.2e-16
```

Note that the coefficient $\beta_2 > 0$ associated with the term $I(Assign)^2$ indicates an increase that starts slowly and 'accelerates'(加速) as Assign increases.

5. Linear models with a categorical (factor) explanatory variable

本节需要的包：

```
require(s20x)
```

```
Loading required package: s20x
```

5.1. Using categorical variables as explanatory variables by using indicator variables 使用指标变量将分类变量用作解释变量

License

Skip to main content

> **Note** This website is built using Nextra, a modern static website generator.