

STAT 359 - Assignment 4

Isabella Pelletier

11/24/2020

Question 1.

Read the data into R, and use `table()` function to produce a contingency table summarizing these data.

```
cancer = read.table(file="~/Desktop/R/LungCancer.csv", header = TRUE, sep=","
)
cancer.table = as.matrix(table(cancer))
cancer.table

##      Smoker
## Case    0    1
##    0  60 650
##    1  22 687
```

- The null hypothesis states that there is no association between smoking and lung cancer.
- The alternative hypothesis states that there is an association between smoking and lung cancer.

Assuming that there is no association between smoking and lung cancer, compute a table of 'expected' counts.

```
chisq.test(cancer.table, correct=FALSE)$expected

##      Smoker
## Case    0    1
##    0 41.02889 668.9711
##    1 40.97111 668.0289
```

By hand, compute the observed value of the test statistic for testing association between lung cancer and smoking.

- To find the observed value of the test statistic under independence, we need to use this equation:

$$\chi^2 = \sum (O_{ij} - E_{ij})^2 / E_{ij}$$

- $\chi^2 = (60-41.02889)^2/41.02889 + (650-668.9711)^2/668.9711 + (22-40.97111)^2/40.97111 + (687-668.0289)^2/668.0289$
- $\chi^2 = 18.63$

Assuming there is no association, what is the distribution of the test statistic?

- To determine if the statistic is 'large' we need to know the null distribution.
- If the null hypothesis is true, then this statistic has degrees of freedom of $(r-1)(c-1)$, which in our case is $(2-1)(2-1)$. This gives us a value of 1.
- So we compare to a χ^2_1 distribution. (chi-square with superscript 2, and subscript 1)

Using R, compute the p-value for a test of association, and give a detailed conclusion based on the p-value and a comparison of the tables observed and expected counts.

```
chisq.test(cancer.table, correct=FALSE)

##
## Pearson's Chi-squared test
##
## data: cancer.table
## X-squared = 18.633, df = 1, p-value = 1.585e-05
```

- We obtained a p-value of 1.585e-05 which provides very strong evidence against the null hypothesis. Thus we reject the null hypothesis of independence and conclude that there is an association between smoking and lung cancer.
- For those who smoke, the number of observed individuals with lung cancer (687) was greater than that expected under independence (668.0289). The observed number of individuals without lung cancer (650) was less than that expected under independence (668.9711).
- For those who do not smoke, the observed number of individuals with lung cancer (22) was less than that expected under independence (40.97111). The observed

number of individuals without lung cancer (60) was greater than that expected under independence (41.02889).

Question 2.

Assuming that there is no association between disease and blood group, compute a table of 'expected' counts.

```
eskimo = matrix(c(7,27,55,7,34,52,7,12,11,13,18,24),nrow=3,ncol=4,dimnames=list(c("Moderately-advanced","Minimal","Not Present"),c("O","A","AB","B")))
eskimo
```

	O	A	AB	B
Moderately-advanced	7	7	7	13
Minimal	27	34	12	18
Not Present	55	52	11	24

```
chisq.test(eskimo,correct=FALSE)$expected
```

```
## Warning in chisq.test(eskimo, correct = FALSE): Chi-squared approximation
may be
## incorrect
```

	O	A	AB	B
Moderately-advanced	11.33333	11.84270	3.820225	7.003745
Minimal	30.33333	31.69663	10.224719	18.745318
Not Present	47.33333	49.46067	15.955056	29.250936

- The null hypothesis states that there is no association between tuberculosis and blood group.
- The alternative hypothesis states that there is an association between tuberculosis and blood group.

By hand, compute the observed value of the test statistic for testing association between disease and blood group.

- To find the observed value of the test statistic under independence, we need to use this equation:

$$\chi^2 = \sum (O_{ij} - E_{ij})^2 / E_{ij}$$

- $\chi^2 = (7-11.33)^2/11.33 + (7-11.8427)^2/11.8427 + (7-3.820225)^2/3.820225 + (13-7.003745)^2/7.003745 + (27-30.33)^2/30.33 + (34-31.69663)^2/31.69663 + (12-10.224719)^2/10.224719 + (18-18.745318)^2/18.745318 + (55-47.33)^2/47.33 + (52-49.46067)^2/49.46067 + (11-15.955056)^2/15.955056 + (24-29.250936)^2/29.250936$
- $\chi^2 = 16.145$

Assuming there is no association, what is the distribution of the test statistic?

- If the null hypothesis is true, then this statistic has degrees of freedom of $(r-1)(c-1)$, which in our case is $(3-1)(4-1)$. This gives us a value of 6.
- So we compare to a χ^2_6 distribution. (chi-square with superscript 2, and subscript 6)

Using R, compute the p-value for a test of association, and give a detailed conclusion based on the p-value and a comparison of the tables observed and expected counts.

```
chisq.test(eskimo, correct=FALSE)
```

```
## Warning in chisq.test(eskimo, correct = FALSE): Chi-squared approximation
## may be
## incorrect

##
## Pearson's Chi-squared test
##
## data:  eskimo
## X-squared = 16.143, df = 6, p-value = 0.01301
```

- We obtained a p-value of 0.01301 which provides minimal evidence against the null hypothesis. Thus we reject the null hypothesis of independence and conclude that there is an association between tuberculosis and blood group.
- For those with blood type O, the number of observed individuals with Moderate-advanced tuberculosis (7) was less than that expected under independence (11.33333). The observed number of individuals with minimal tuberculosis (27) was less than that expected under independence (30.33333). The observed number of individuals with no presence of tuberculosis (55) was greater than expected under independence (47.33333).
- For those with blood type A, the number of observed individuals with Moderate-advanced tuberculosis (7) was less than that expected under independence (11.84270). The observed number of individuals with minimal tuberculosis (34) was

greater than that expected under independence (31.69663). The observed number of individuals with no presence of tuberculosis (52) was greater than expected under independence (49.46067).

- For those with blood type AB, the number of observed individuals with Moderate-advanced tuberculosis (7) was greater than that expected under independence (3.820225). The observed number of individuals with minimal tuberculosis (12) was greater than that expected under independence (10.224719). The observed number of individuals with no presence of tuberculosis (11) was less than expected under independence (15.955056).
- For those with blood type B, the number of observed individuals with Moderate-advanced tuberculosis (13) was greater than that expected under independence (7.003745). The observed number of individuals with minimal tuberculosis (18) was relatively the same of that expected under independence (18.745318). The observed number of individuals with no presence of tuberculosis (24) was less than expected under independence (29.250936).

Question 3.

(a) Produce 4 scatter plots (one for each data set), on the same page, illustrating the relationship between Y and X. Describe each of these briefly, and state if you think a linear model of the form $y_i = a + b x_i + \epsilon_i$ (simple linear regression) would be appropriate.

```
set1x<-c(10,8,13,9,11,14,6,4,12,7,5)
set1y<-c(8.04,6.95,7.58,8.81,8.33,9.96,7.24,4.26,10.84,4.82,5.68)
dataset1 = cbind(set1x,set1y)

set2x<-c(10,8,13,9,11,14,6,4,12,7,5)
set2y<-c(9.14,8.14,8.74,8.77,9.26,8.1,6.13,3.1,9.13,7.26,4.74)
dataset2 = cbind(set2x,set2y)

set3x<-c(10,8,13,9,11,14,6,4,12,7,5)
set3y<-c(7.46,6.77,12.74,7.11,7.81,8.84,6.08,5.39,8.15,6.42,5.73)
dataset3 = cbind(set3x,set3y)

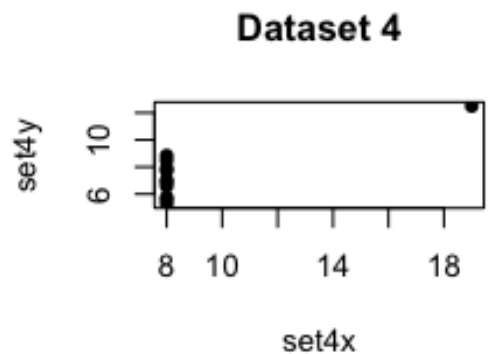
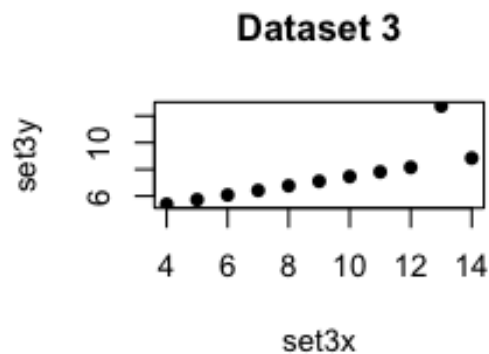
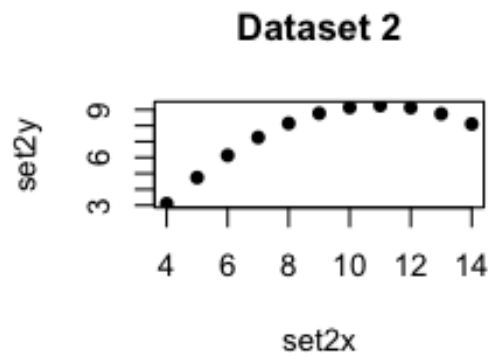
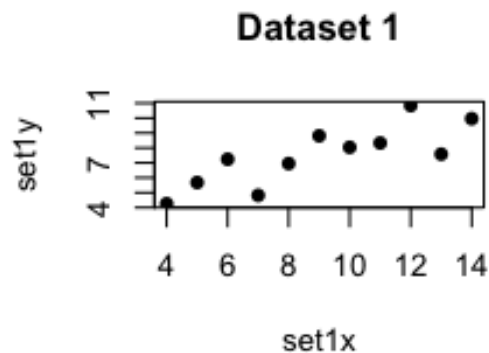
set4x<-c(8,8,8,8,8,8,8,19,8,8,8)
set4y<-c(6.58,5.76,7.71,8.84,8.47,7.04,5.25,12.5,5.56,7.91,6.89)
dataset4 = cbind(set4x,set4y)

par(mfrow=c(2,2))
plot(dataset1, pch=16)
```

```

title("Dataset 1")
plot(dataset2, pch=16)
title("Dataset 2")
plot(dataset3, pch=16)
title("Dataset 3")
plot(dataset4, pch=16)
title("Dataset 4")

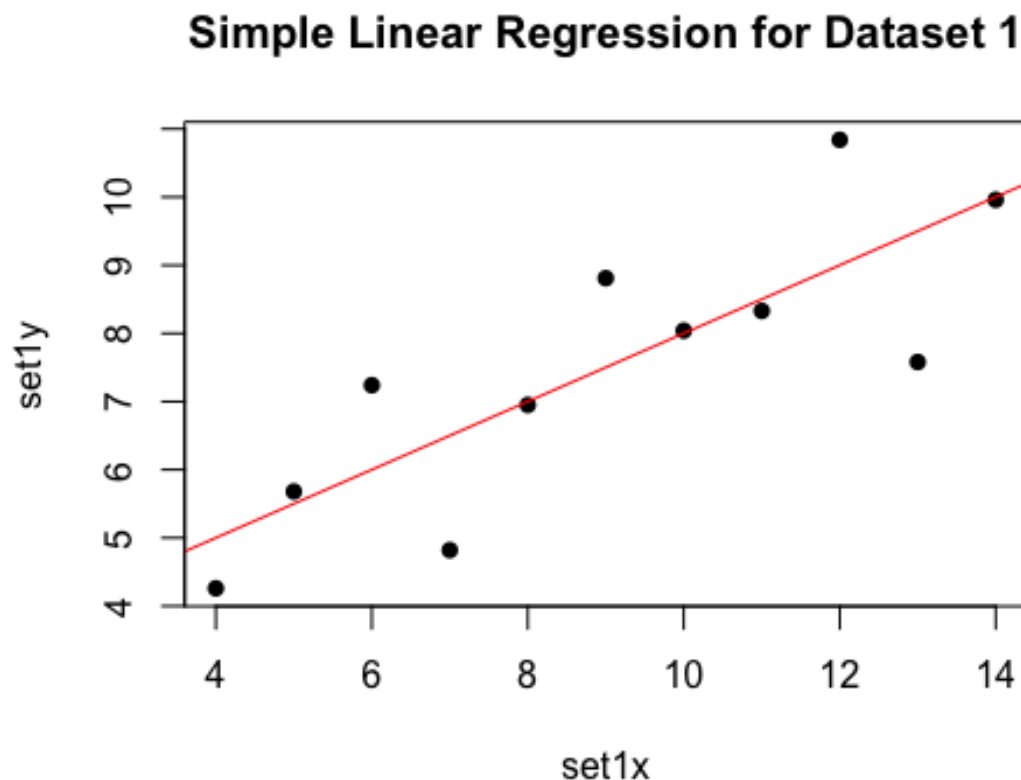
```



- Dataset 1 has a weak linear relationship, and a positive association and correlation between X and Y. A linear regression model could be appropriate.
- Dataset 2 seems to be more quadratic than linear, but there is a positive correlation between X and Y. A simple linear regression model may fit but a non-linear model would be more appropriate.
- Dataset 3 has a stronger linear relationship, and a positive correlation between X and Y but there is an outlier at X=13. A non-linear model would be more appropriate than a linear one.
- Dataset 4 is not linear with no direction. There is an outlier at X=19. A non-linear model would be more appropriate than a linear one.

(b) Perform 4 separate simple linear regressions (one for each data set) and produce a table (in your text editor (ie. word)) that shows the R-squared value. Discuss what is happening here (hint: for simple linear regression, R-squared is just the square of the sample correlation coefficient).

```
plot(dataset1, main="Simple Linear Regression for Dataset 1", pch=16)  
abline(lm(set1y~set1x), col="red")
```

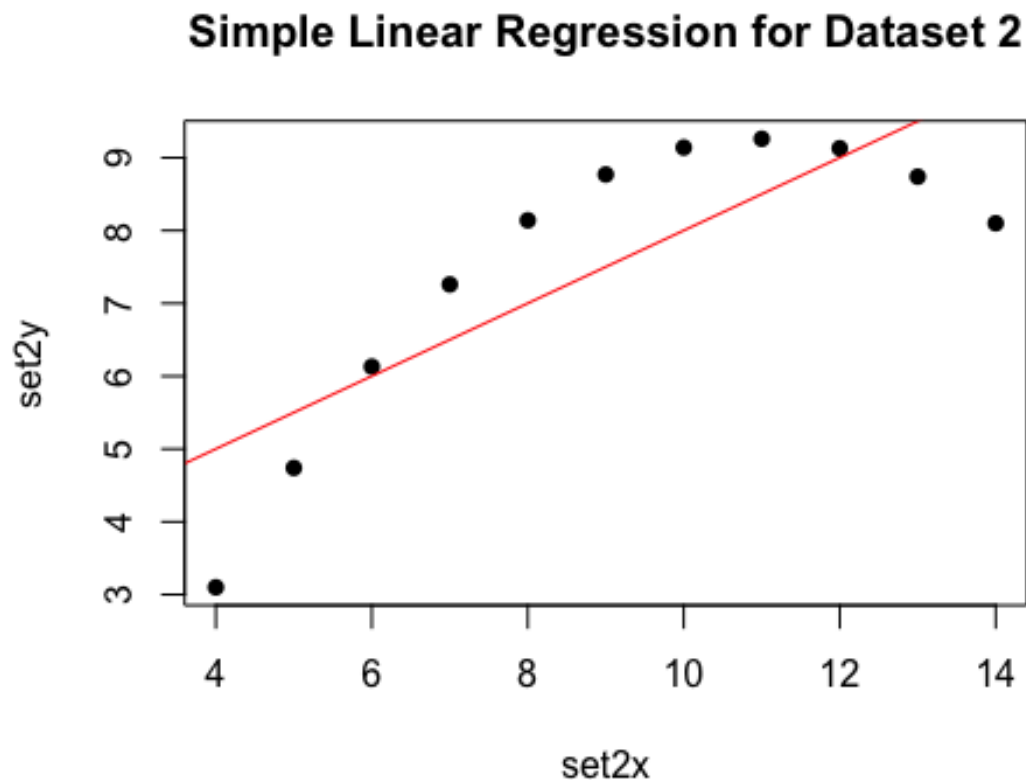


```
summary(lm(set1y~set1x))  
  
##  
## Call:  
## lm(formula = set1y ~ set1x)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max   
## -1.92127 -0.45577 -0.04136  0.70941  1.83882   
##  
## Coefficients:  
##              Estimate Std. Error t value Pr(>|t|)      
## (Intercept)   3.0001     1.1247   2.667  0.02573 *  
##
```

```
## set1x          0.5001      0.1179   4.241  0.00217 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.237 on 9 degrees of freedom
## Multiple R-squared:  0.6665, Adjusted R-squared:  0.6295
## F-statistic: 17.99 on 1 and 9 DF,  p-value: 0.00217
```

- For Dataset 1 the R-squared value is 0.6665, which means that 66.65% of the variation in Y can be explained by X. The value is positive which demonstrates positive correlation between X and Y. #

```
plot(dataset2, main="Simple Linear Regression for Dataset 2", pch=16)
abline(lm(set2y~set2x), col="red")
```



```
summary(lm(set2y~set2x))

##
## Call:
## lm(formula = set2y ~ set2x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.9009 -0.7609  0.1291  0.9491  1.2691
```

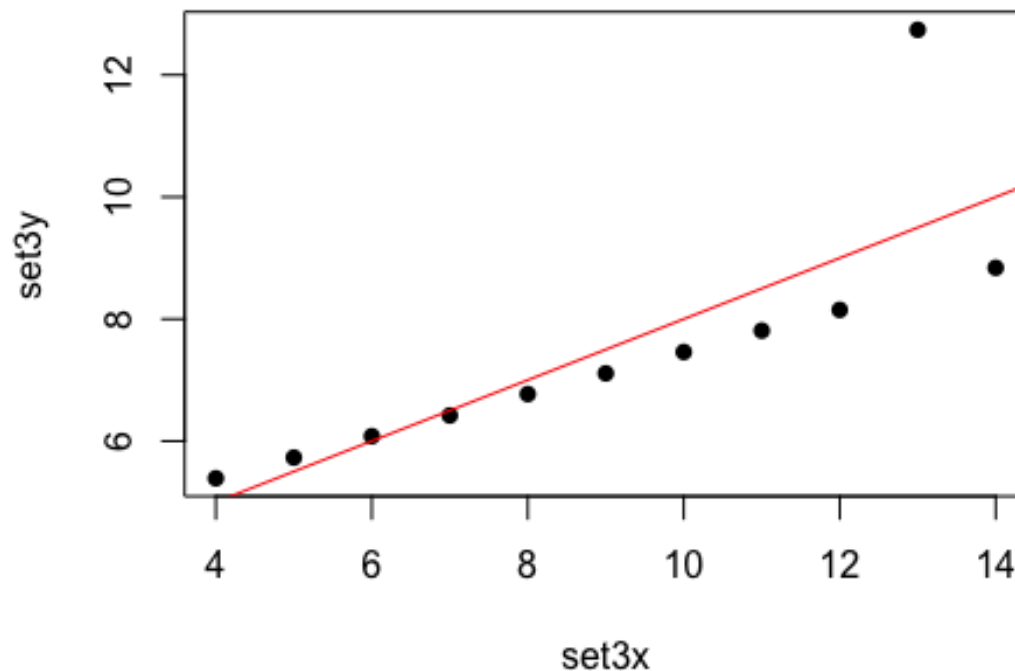


```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.001      1.125   2.667  0.02576 *
## set2x         0.500      0.118   4.239  0.00218 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.237 on 9 degrees of freedom
## Multiple R-squared:  0.6662, Adjusted R-squared:  0.6292
## F-statistic: 17.97 on 1 and 9 DF,  p-value: 0.002179
```

- For Dataset 2 the R-squared value is 0.6662, which means that 66.62% of the variation in Y can be explained by X. The value is positive which demonstrates positive correlation between X and Y. #

```
plot(dataset3, main="Simple Linear Regression for Dataset 3", pch=16)
abline(lm(set3y~set3x), col="red")
```

Simple Linear Regression for Dataset 3



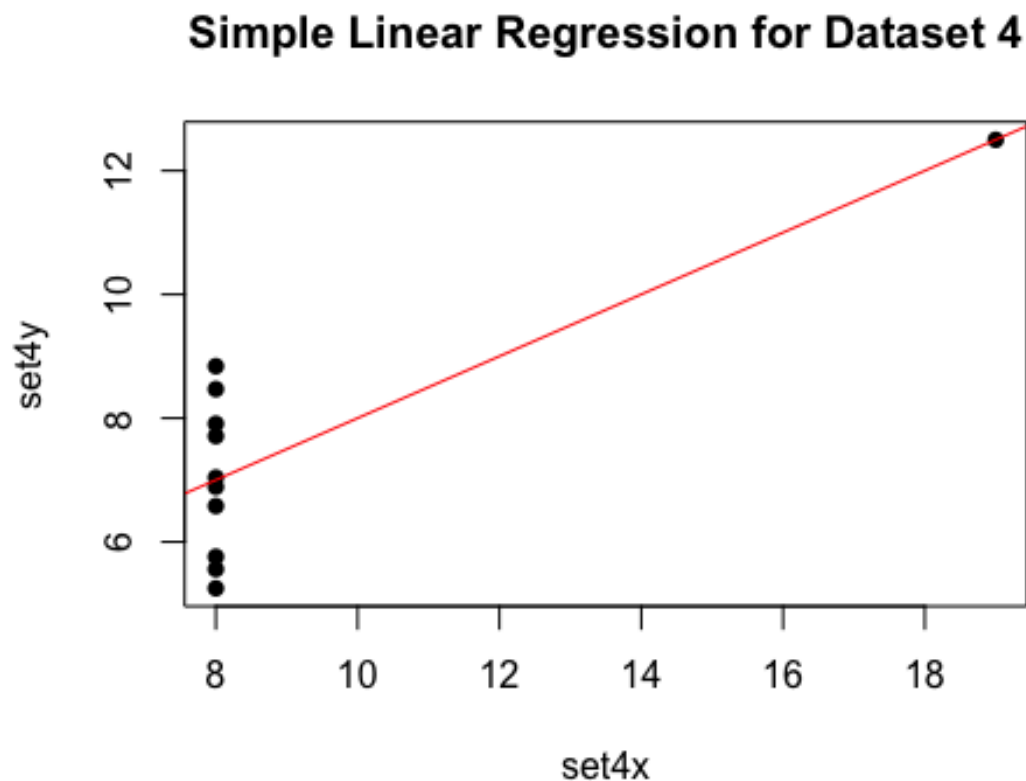
```
summary(lm(set3y~set3x))

##
## Call:
## lm(formula = set3y ~ set3x)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.1586 -0.6146 -0.2303  0.1540  3.2411
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0025     1.1245   2.670  0.02562 *
## set3x         0.4997     0.1179   4.239  0.00218 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.236 on 9 degrees of freedom
## Multiple R-squared:  0.6663, Adjusted R-squared:  0.6292
## F-statistic: 17.97 on 1 and 9 DF,  p-value: 0.002176
```

- For Dataset 3 the R-squared value is 0.6663, which means that 66.63% of the variation in Y can be explained by X. The value is positive which demonstrates positive correlation between X and Y. #

```
plot(dataset4, main="Simple Linear Regression for Dataset 4", pch=16)
abline(lm(set4y~set4x), col="red")
```



```
summary(lm(set4y~set4x))
```

```
##
## Call:
## lm(formula = set4y ~ set4x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.751 -0.831  0.000  0.809  1.839
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0017     1.1239   2.671  0.02559 *
## set4x         0.4999     0.1178   4.243  0.00216 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.236 on 9 degrees of freedom
## Multiple R-squared:  0.6667, Adjusted R-squared:  0.6297
## F-statistic:    18 on 1 and 9 DF,  p-value: 0.002165
```

- For Dataset 4 the R-squared value is 0.6667, which means that 66.67% of the variation in Y can be explained by X. The value is positive which demonstrates positive correlation between X and Y. #

```
table<- matrix(c(0.6665,0.6662,0.6663,0.6667),nrow=4,ncol=1,dimnames=list(c("
Dataset 1", "Dataset 2", "Dataset 3", "Dataset 4"),("R-squared Value")))
table
```

```
##           R-squared Value
## Dataset 1           0.6665
## Dataset 2           0.6662
## Dataset 3           0.6663
## Dataset 4           0.6667
```

- It is apparent from the graphs and summaries that although the Datasets differ, they all have approximately the same linear regression line, fit of model, and a positive correlation.

Question 4.

The file 'growth' gives data on the height of a white spruce tree measured annually for 50 years. Letting Y_t denote the height of the tree at year $t > 0$, we consider describing the growth of the tree over time with a non-linear model $Y_t = f(t) + \epsilon_t$, $\epsilon_t \sim N(0, \sigma^2)$. Three growth curves are considered for $f(t)$. __ (a) Logistic: $f(t) =$

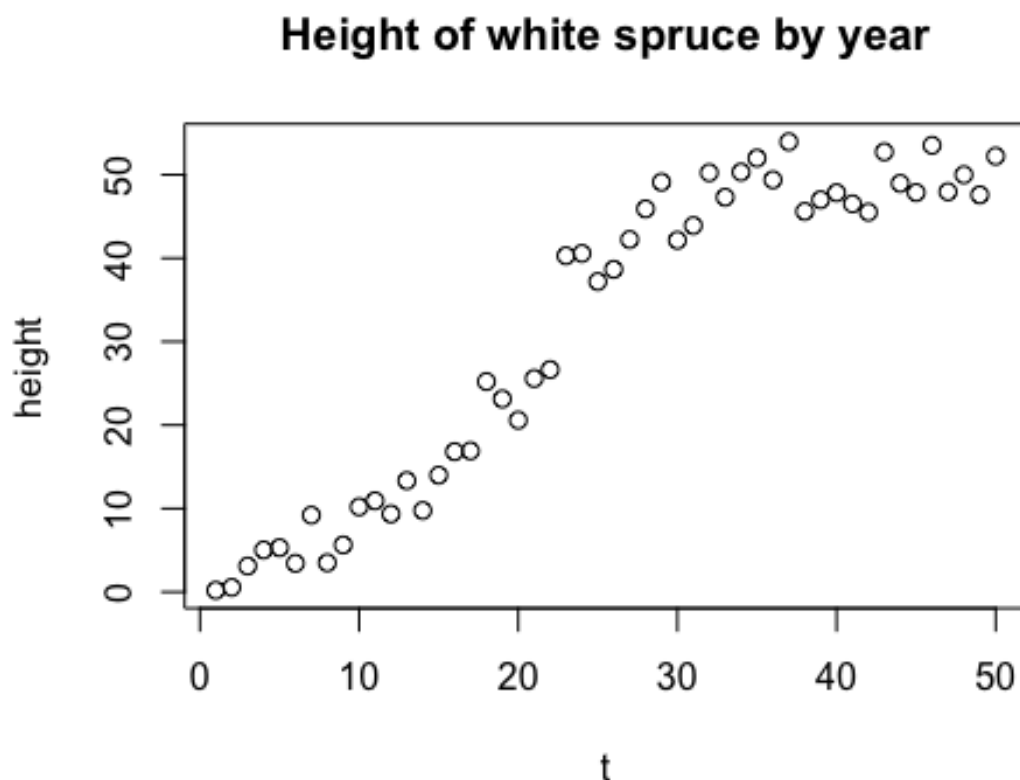
$a/(1+b \cdot \exp\{-ct\})$ (b) Gompertz: $f(t) = a \exp\{-b \exp\{-ct\}\}$ (c) Von Bertalanffy: $f(t) = a - a \exp\{-b(t+c)\}$ __

```
growth = read.table(file="~/Desktop/R/growth.txt", header=TRUE, sep="")
attach(growth)
names(growth)

## [1] "t"      "height"
```

Fit all three models using the non-linear least squares function `nls()` in R. Explain how you are choosing the starting values for `nls()` in each case. Produce a figure depicting the estimated curves all on the same plot, along with the observed data. Be sure to include a legend to distinguish the different curves.

```
plot(t,height)
title("Height of white spruce by year")
```



- For all three models we will have to calculate and select the starting values and evaluate $f(t)$ at $t=0$, and $t=\infty$.
- The hypothesis test for all three models are generally the same. $H0a: a=0$, $H0b: b=0$, $H0c: c=0$ and $H1a: a\neq 0$, $H1b: b\neq 0$, $H1c: c\neq 0$

```
library(nls2)

## Loading required package: proto

# For model 1 at t=0, f(t) = a/(1+b)
# at t=∞, f(t) = a
# we know that a is the asymptote so from the horizontal asymptote on the plot a = 52
# at 0.02 there is a y-intercept which gives us 0.02 = 52/(1+b), solving for b gives us b = 259
# now, solving for c at the steepest part of the plot y=22 and t=19, so c=(-1/19)*ln((52/(259*22))-(1/259))=0.276
a1 = 52
b1 = 259
c1 = 0.276
model1 = nls(height~(a1/(1+b1*exp(-c1*t))), start = list(a1=52,b1=259,c1=0.276))
summary(model1)

##
## Formula: height ~ (a1/(1 + b1 * exp(-c1 * t)))
##
## Parameters:
##      Estimate Std. Error t value Pr(>|t|)
## a1   50.4207      0.8473  59.509  < 2e-16 ***
## b1   47.1377     12.1605   3.876  0.000328 ***
## c1    0.1993      0.0141  14.132  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.069 on 47 degrees of freedom
##
## Number of iterations to convergence: 9
## Achieved convergence tolerance: 1.609e-06
```

- The p-values for all of $a1$, $b1$, and $c1$ are all less than 0.05 and thus are significant. All null hypotheses can be rejected for model 1.

```
# For model 2 at t=0, f(t) = a*exp(-b)
# at t=∞, f(t) = a
# we know that a is the asymptote so from the horizontal asymptote on the plot a = 52
```

```

# at 0.02 there is a y-intercept which gives us  $0.02 = 52 \cdot \exp(-b)$ , solving for b gives us  $b = 7.86$ 
# now, solving for c at the steepest part of the plot  $y=22$  and  $t=19$ , so  $c = (-1/19) \cdot (\ln((-1/7.86) \cdot \ln(22/52))) = 0.116$ 
a2=52
b2=7.86
c2=0.116
model2 = nls(height~(a2*exp(-b2*exp(-c2*t))), start=list(a2=52, b2=7.86, c2=0.116))
summary(model2)

##
## Formula: height ~ (a2 * exp(-b2 * exp(-c2 * t)))
##
## Parameters:
##      Estimate Std. Error t value Pr(>|t|)
## a2 52.21786    1.33361  39.155 < 2e-16 ***
## b2  7.57920    1.40076   5.411 2.07e-06 ***
## c2  0.12491    0.01156  10.802 2.50e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.617 on 47 degrees of freedom
##
## Number of iterations to convergence: 8
## Achieved convergence tolerance: 5.194e-06

```

- The p-values for all of a2, b2, and c2 are all less than 0.05 and thus are significant. All null hypotheses can be rejected for model 2.

```

# For model 3 at  $t=0$ ,  $f(t) = a - a \cdot \exp(-b \cdot c)$ 
# at  $t=\infty$ ,  $f(t) = a$ 
# we know that a is the asymptote so from the horizontal asymptote on the plot  $a = 52$ 
# at 0.02 there is a y-intercept which gives us  $0.02 = 52 - 52 \cdot \exp(-b \cdot c)$ 
# there are two unknown values in this equation so we need another one.
# using the steepest part of the plot we get  $c = ((-1/b) \cdot \ln(1 - (22/52))) - 19$ 
# then using  $b = (-1/((( -1/b) \cdot \ln(1 - (22/52))) - 19)) \cdot \ln(1 - (0.02/52))$  subbing in our equation for c we get  $b = 0.0289$ 
# subbing our new b into the c expression we get  $c = 0.03275$ 
a3=52
b3=0.0289
c3=0.03275
model3 = nls(height~(a3-a3*exp(-b3*(t+c3))), start=list(a3=52,b3=0.0289,c3=0.03275))
summary(model3)

##
## Formula: height ~ (a3 - a3 * exp(-b3 * (t + c3)))
##
## Parameters:

```

```
##      Estimate Std. Error t value Pr(>|t|)
## a3 74.411730    9.950840   7.478 1.55e-09 ***
## b3  0.029390    0.006914   4.251  1e-04 ***
## c3 -3.760728    0.834246  -4.508 4.34e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.366 on 47 degrees of freedom
##
## Number of iterations to convergence: 7
## Achieved convergence tolerance: 7.334e-06
```

- The p-values for all of a3, b3, and c3 are all less than 0.05 and thus are significant. All null hypotheses can be rejected for model 3.

```
plot(t,height, pch=16)
t.fit = seq(0,50,0.1)
model1.fit = predict(model1, list(t=t.fit))
model2.fit = predict(model2, list(t=t.fit))
model3.fit = predict(model3, list(t=t.fit))
lines(t.fit,model1.fit, col="black", lty=2)
lines(t.fit,model2.fit, col="red", lty=2)
lines(t.fit,model3.fit, col="blue", lty=2)
title("Height of White Spruce by Year")
legend

## function (x, y = NULL, legend, fill = NULL, col = par("col"),
##      border = "black", lty, lwd, pch, angle = 45, density = NULL,
##      bty = "o", bg = par("bg"), box.lwd = par("lwd"), box.lty = par("lty"),
##      box.col = par("fg"), pt.bg = NA, cex = 1, pt.cex = cex, pt.lwd = lwd,
##      xjust = 0, yjust = 1, x.intersp = 1, y.intersp = 1, adj = c(0,
##      0.5), text.width = NULL, text.col = par("col"), text.font = NULL,
##      merge = do.lines && has.pch, trace = FALSE, plot = TRUE,
##      ncol = 1, horiz = FALSE, title = NULL, inset = 0, xpd, title.col = tex
t.col,
##      title.adj = 0.5, seg.len = 2)
## {
##   if (missing(legend) && !missing(y) && (is.character(y) ||
##       is.expression(y))) {
##     legend <- y
##     y <- NULL
##   }
##   mfill <- !missing(fill) || !missing(density)
##   if (!missing(xpd)) {
##     op <- par("xpd")
##     on.exit(par(xpd = op))
##     par(xpd = xpd)
##   }
##   title <- as.graphicsAnnot(title)
##   if (length(title) > 1)
##     stop("invalid 'title'")
```

```

## legend <- as.graphicsAnnot(legend)
## n.leg <- if (is.call(legend))
##     1
## else length(legend)
## if (n.leg == 0)
##     stop("'legend' is of length 0")
## auto <- if (is.character(x))
##     match.arg(x, c("bottomright", "bottom", "bottomleft",
##         "left", "topleft", "top", "topright", "right", "center"))
## else NA
## if (is.na(auto)) {
##     xy <- xy.coords(x, y, setLab = FALSE)
##     x <- xy$x
##     y <- xy$y
##     nx <- length(x)
##     if (nx < 1 || nx > 2)
##         stop("invalid coordinate lengths")
## }
## else nx <- 0
## xlog <- par("xlog")
## ylog <- par("ylog")
## rect2 <- function(left, top, dx, dy, density = NULL, angle,
##     ...) {
##     r <- left + dx
##     if (xlog) {
##         left <- 10^left
##         r <- 10^r
##     }
##     b <- top - dy
##     if (ylog) {
##         top <- 10^top
##         b <- 10^b
##     }
##     rect(left, top, r, b, angle = angle, density = density,
##         ...)
## }
## segments2 <- function(x1, y1, dx, dy, ...) {
##     x2 <- x1 + dx
##     if (xlog) {
##         x1 <- 10^x1
##         x2 <- 10^x2
##     }
##     y2 <- y1 + dy
##     if (ylog) {
##         y1 <- 10^y1
##         y2 <- 10^y2
##     }
##     segments(x1, y1, x2, y2, ...)
## }
## points2 <- function(x, y, ...) {

```



```

##         if (xlog)
##             x <- 10^x
##         if (ylog)
##             y <- 10^y
##         points(x, y, ...)
##     }
##     text2 <- function(x, y, ...) {
##         if (xlog)
##             x <- 10^x
##         if (ylog)
##             y <- 10^y
##         text(x, y, ...)
##     }
##     if (trace) {
##         catn <- function(...) do.call(cat, c(lapply(list(...),
##             formatC), "\n"))
##         fv <- function(...) paste(vapply(lapply(list(...), formatC),
##             paste, collapse = ",", ""), collapse = ", ")
##     }
##     Cex <- cex * par("cex")
##     if (is.null(text.width))
##         text.width <- max(abs(strwidth(legend, units = "user",
##             cex = cex, font = text.font)))
##     else if (!is.numeric(text.width) || text.width < 0)
##         stop("'text.width' must be numeric, >= 0")
##     xyc <- xyinch(par("cin"), warn.log = FALSE)
##     xc <- Cex * xyc[1L]
##     yc <- Cex * xyc[2L]
##     if (any(n_ <- xc < 0))
##         text.width[n_] <- -text.width[n_]
##     xchar <- xc
##     xextra <- 0
##     yextra <- yc * (y.intersp - 1)
##     ymax <- yc * max(1, strheight(legend, units = "user", cex = cex)/yc)
##     ychar <- yextra + ymax
##     if (trace)
##         catn(" xchar=", fv(xchar), "; (yextra, ychar)=", fv(yextra,
##             ychar))
##     if (mfill) {
##         xbox <- xc * 0.8
##         ybox <- yc * 0.5
##         dx.fill <- max(xbox)
##     }
##     do.lines <- (!missing(lty) && (is.character(lty) || any(lty >
##         0))) || !missing(lwd)
##     n.legpercol <- if (horiz) {
##         if (ncol != 1)
##             warning(gettextf("horizontal specification overrides: Number o
## f columns := %d",
##                 n.leg), domain = NA)

```

```

##         ncol <- n.leg
##         1
##     }
##     else ceiling(n.leg/ncol)
##     has.pch <- !missing(pch) && length(pch) > 0
##     if (do.lines) {
##         x.off <- if (merge)
##             -0.7
##         else 0
##     }
##     else if (merge)
##         warning("'merge = TRUE' has no effect when no line segments are drawn")
##     if (has.pch) {
##         if (is.character(pch) && !is.na(pch[1L]) && nchar(pch[1L],
##             type = "c") > 1) {
##             if (length(pch) > 1)
##                 warning("not using pch[2..] since pch[1L] has multiple characters")
##             np <- nchar(pch[1L], type = "c")
##             pch <- substr(rep.int(pch[1L], np), 1L:np, 1L:np)
##         }
##         if (!is.character(pch))
##             pch <- as.integer(pch)
##     }
##     if (is.na(auto)) {
##         if (xlog)
##             x <- log10(x)
##         if (ylog)
##             y <- log10(y)
##     }
##     if (nx == 2) {
##         x <- sort(x)
##         y <- sort(y)
##         left <- x[1L]
##         top <- y[2L]
##         w <- diff(x)
##         h <- diff(y)
##         w0 <- w/ncol
##         x <- mean(x)
##         y <- mean(y)
##         if (missing(xjust))
##             xjust <- 0.5
##         if (missing(yjust))
##             yjust <- 0.5
##     }
##     else {
##         h <- (n.legpercol + !is.null(title)) * ychar + yc
##         xch1 <- max(xchar)
##         w0 <- text.width + (x.intersp + 1) * xch1

```

```

##      if (mfill)
##          w0 <- w0 + dx.fill
##      if (do.lines)
##          w0 <- w0 + (seg.len + x.off) * xch1
##      w <- ncol * w0 + 0.5 * xch1
##      if (!is.null(title) && (abs(tw <- strwidth(title, units = "user",
##          cex = cex) + 0.5 * xchar)) > abs(w)) {
##          xextra <- (tw - w)/2
##          w <- tw
##      }
##      if (is.na(auto)) {
##          left <- x - xjust * w
##          top <- y + (1 - yjust) * h
##      }
##      else {
##          usr <- par("usr")
##          inset <- rep_len(inset, 2)
##          insetx <- inset[1L] * (usr[2L] - usr[1L])
##          left <- switch(auto, bottomright = , topright = ,
##              right = usr[2L] - w - insetx, bottomleft = ,
##              left = , topleft = usr[1L] + insetx, bottom = ,
##              top = , center = (usr[1L] + usr[2L] - w)/2)
##          insety <- inset[2L] * (usr[4L] - usr[3L])
##          top <- switch(auto, bottomright = , bottom = , bottomleft = us
r[3L] +
##              h + insety, topleft = , top = , topright = usr[4L] -
##              insety, left = , right = , center = (usr[3L] +
##              usr[4L] + h)/2)
##      }
##  }
##  if (plot && bty != "n") {
##      if (trace)
##          catn(" rect2(", left, ",", top, ", w=", w, ", h=",
##              h, ", ...)", sep = "")
##      rect2(left, top, dx = w, dy = h, col = bg, density = NULL,
##          lwd = box.lwd, lty = box.lty, border = box.col)
##  }
##  xt <- left + xchar + xextra + (w0 * rep.int(0:(ncol - 1),
##      rep.int(n.legpercol, ncol)))[1L:n.leg]
##  yt <- top - 0.5 * yextra - ymax - (rep.int(1L:n.legpercol,
##      ncol)[1L:n.leg] - 1 + !is.null(title)) * ychar
##  if (mfill) {
##      if (plot) {
##          if (!is.null(fill))
##              fill <- rep_len(fill, n.leg)
##          rect2(left = xt, top = yt + ybox/2, dx = xbox, dy = ybox,
##              col = fill, density = density, angle = angle,
##              border = border)
##      }
##      xt <- xt + dx.fill

```

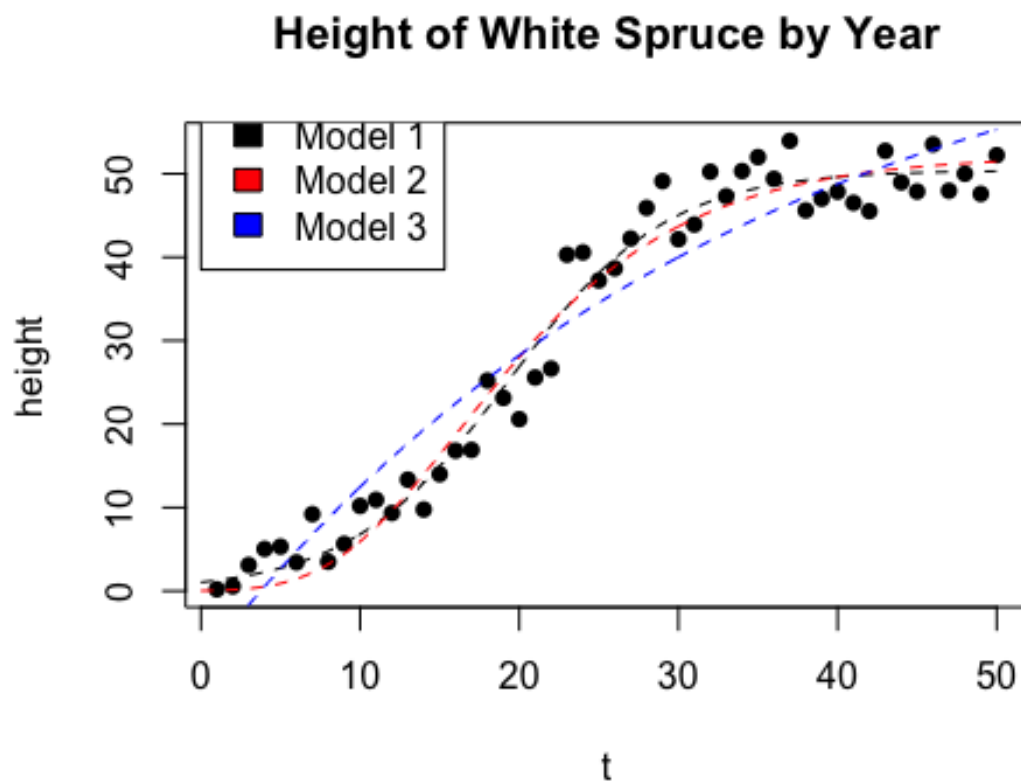
```

## }
## if (plot && (has.pch || do.lines))
##   col <- rep_len(col, n.leg)
## if (missing(lwd) || is.null(lwd))
##   lwd <- par("lwd")
## if (do.lines) {
##   if (missing(lty) || is.null(lty))
##     lty <- 1
##   lty <- rep_len(lty, n.leg)
##   lwd <- rep_len(lwd, n.leg)
##   ok.l <- !is.na(lty) & (is.character(lty) | lty > 0) &
##     !is.na(lwd)
##   if (trace)
##     catn(" segments2(", xt[ok.l] + x.off * xchar, ",",
##         yt[ok.l], ", dx=", seg.len * xchar, ", dy=0, ...)")
##   if (plot)
##     segments2(xt[ok.l] + x.off * xchar, yt[ok.l], dx = seg.len *
##       xchar, dy = 0, lty = lty[ok.l], lwd = lwd[ok.l],
##       col = col[ok.l])
##   xt <- xt + (seg.len + x.off) * xchar
## }
## if (has.pch) {
##   pch <- rep_len(pch, n.leg)
##   pt.bg <- rep_len(pt.bg, n.leg)
##   pt.cex <- rep_len(pt.cex, n.leg)
##   pt.lwd <- rep_len(pt.lwd, n.leg)
##   ok <- !is.na(pch)
##   if (!is.character(pch)) {
##     ok <- ok & (pch >= 0 | pch <= -32)
##   }
##   else {
##     ok <- ok & nzchar(pch)
##   }
##   x1 <- (if (merge && do.lines)
##     xt - (seg.len/2) * xchar
##   else xt)[ok]
##   y1 <- yt[ok]
##   if (trace)
##     catn(" points2(", x1, ",", y1, ", pch=", pch[ok],
##         ", ...)")
##   if (plot)
##     points2(x1, y1, pch = pch[ok], col = col[ok], cex = pt.cex[ok]
## ,
##       bg = pt.bg[ok], lwd = pt.lwd[ok])
## }
## xt <- xt + x.intersp * xchar
## if (plot) {
##   if (!is.null(title))
##     text2(left + w * title.adj, top - ymax, labels = title,
##       adj = c(title.adj, 0), cex = cex, col = title.col)

```

```
##      text2(xt, yt, labels = legend, adj = adj, cex = cex,
##          col = text.col, font = text.font)
##    }
##    invisible(list(rect = list(w = w, h = h, left = left, top = top),
##        text = list(x = xt, y = yt)))
## }
## <bytecode: 0x7fbedf012630>
## <environment: namespace:graphics>

legend(x=0,y=60,legend=c("Model 1", "Model 2", "Model 3"), fill=c("black", "red", "blue"))
```



*

Models 1 and two seem to have the best fit.

For each of the three models, give a 95% confidence interval for $\lim_{t \rightarrow \infty} f(t)$, What does this represent?

```
#1
a.CI1 = c(50.4207-1.96*0.8473,50.4207+1.96*0.8473)
a.CI1
```

```
## [1] 48.75999 52.08141
```

```
#2
```

```
a.CI2 = c(52.21786-1.96*1.33361,52.21786+1.96*1.33361)
```

```
a.CI2
```

```
## [1] 49.60398 54.83174
```

```
#3
```

```
a.CI3 = c(74.411730-1.96*9.950840,74.411730+1.96*9.950840)
```

```
a.CI3
```

```
## [1] 54.90808 93.91538
```

- The confidence intervals for a1 and a2 overlap which means that they are similar in function.

Select the best of the three models, and plot an estimate of the derivative $df(t)/dt$, which represents the rate of growth over time.

```
# It seemed as though model 1 was the best fit of the three models.
```

```
plot(t,height, pch=16)
```

```
t.fit = seq(0,50,0.1)
```

```
modell1.fit = predict(modell1, list(t=t.fit))
```

```
lines(t.fit,modell1.fit, col="black", lty=2)
```

```
title("Growth rate over time")
```

Growth rate over time

