Biochemical Skills 1 : Data Analysis

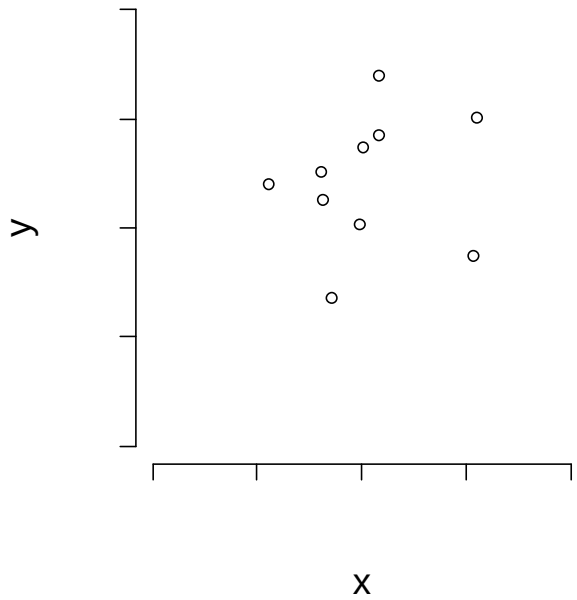
Linear Regression and calibration

Overview of topics

Week	Topic
	Introduction to module, statistics and RStudio including first figure Hypothesis testing, variable types; functions (inbuilt), different ways of getting data into RStudio, getting help in RStudio
	The normal distribution, summary statistics and confidence intervals, RStudio
	One- and two-sample t-tests
	More than two samples: One-way ANOVA
	Linear Regression and Calibration

Summary of this week

- This week we will consider situations where our explanatory variable is more continuous than categorical.



Learning objectives for the week

By actively following the lecture and practical and carrying out the independent study the successful student will be able to:

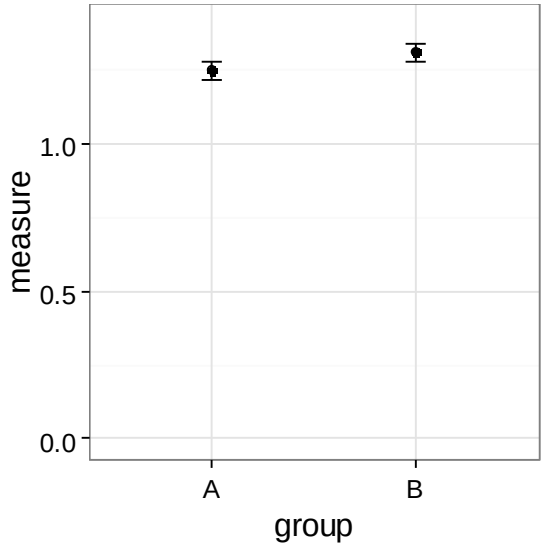
- Explain the rationale behind regression (MLO 2)
- Apply (appropriately), interpret and evaluate the legitimacy of linear regression for data analysis in R (MLO 2 and 3)
- Summarise and illustrate with appropriate R figures test results scientifically (MLO 3)
- Use linear regression to develop and make ‘reverse predictions’ from a calibration (MLO 2 and 3)

Choosing tests

Explanatory variables to explain the response variable.

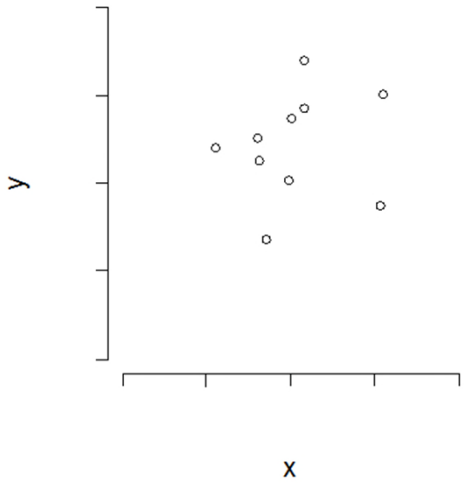
The type of test depends on the type of question and the type of data.

t-tests
ANOVA



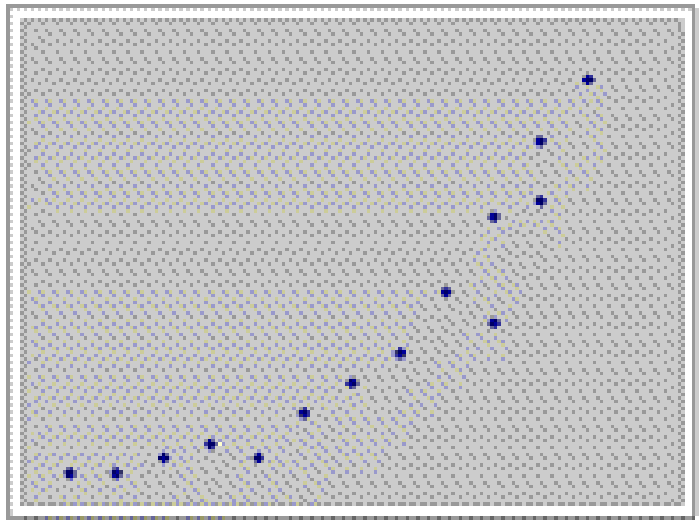
Categorical

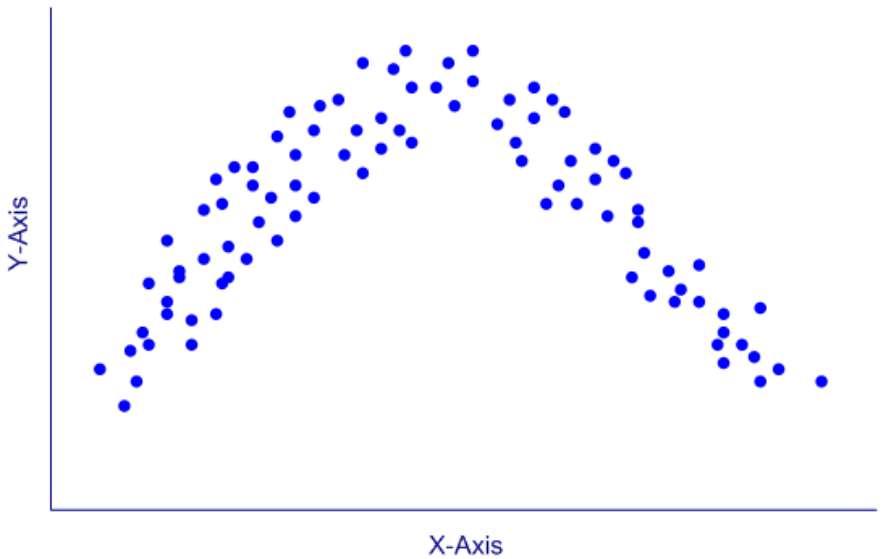
Correlation
Regression

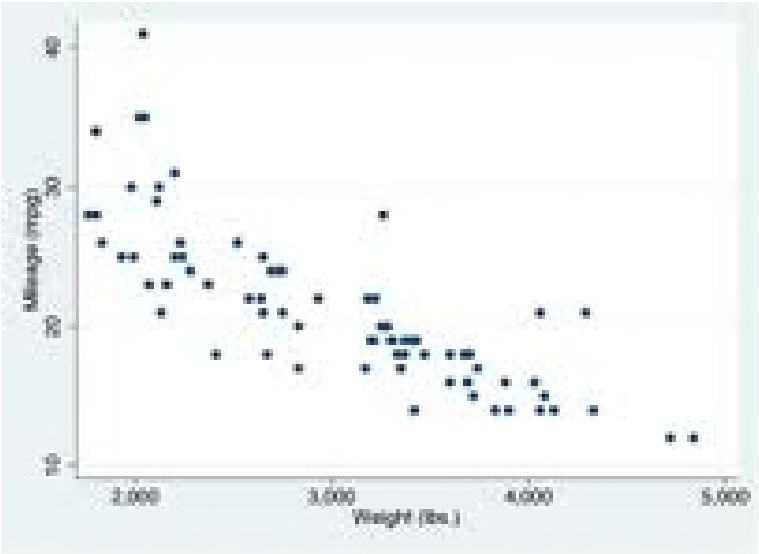


Continuous

Not for linear methods





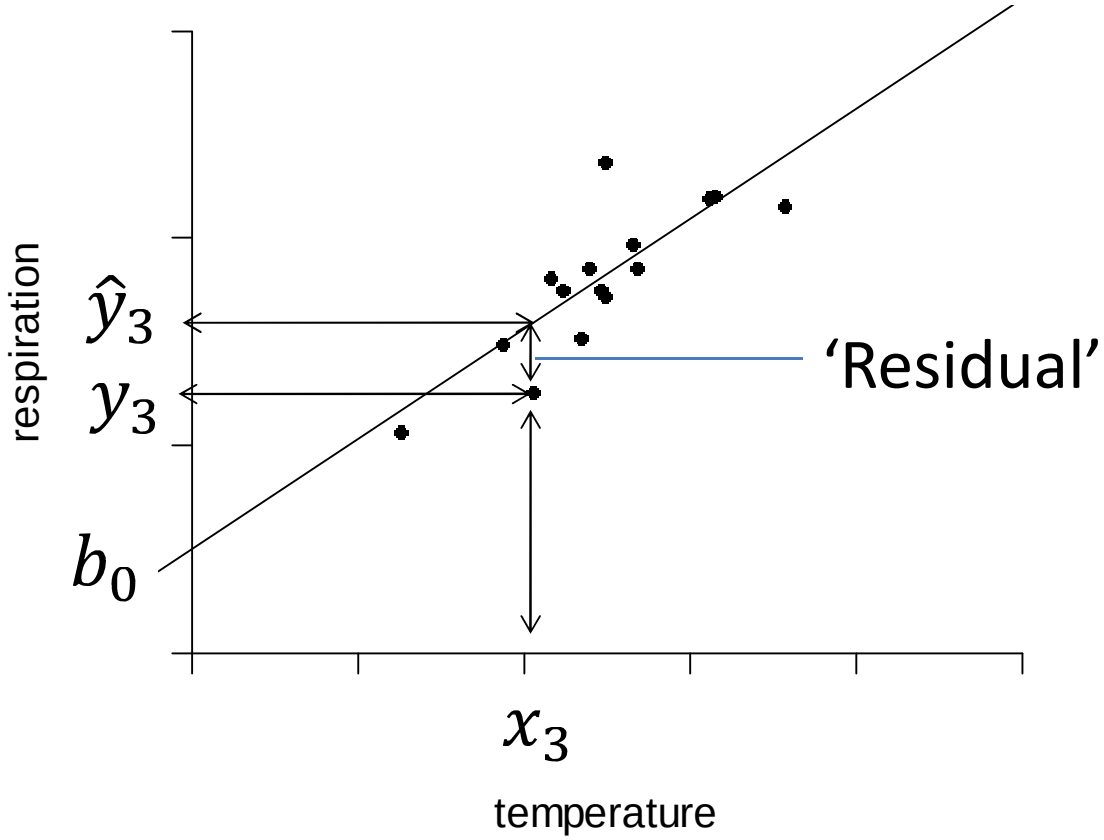


Regression

- Prediction
- One variable causes the other
- Axes matter
- We will consider linear regression only best fitting straight line:

$$y = \underline{b_1} x + \underline{b_0}$$

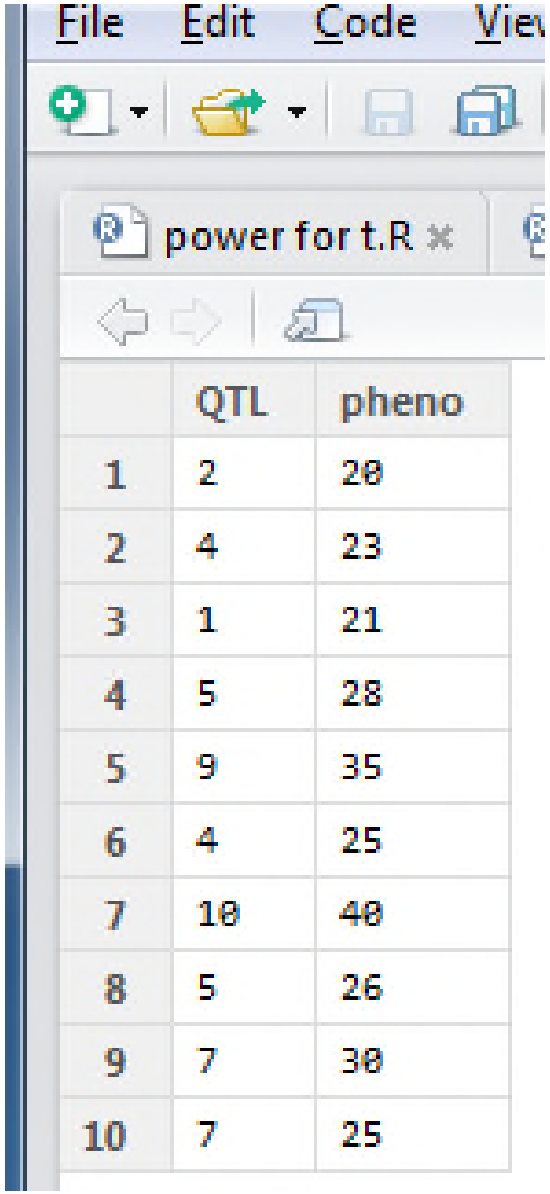
Regression



$$y = b_1 x + b_0$$

Regression

- Null hypothesis can be expressed as:
  - $b_1 = 0$
  - x does not explain *y*
  - Regression line doesn't explain variance in *y*



	QTL	pheno
1	2	20
2	4	23
3	1	21
4	5	28
5	9	35
6	4	25
7	10	40
8	5	26
9	7	30
10	7	25

Regression example

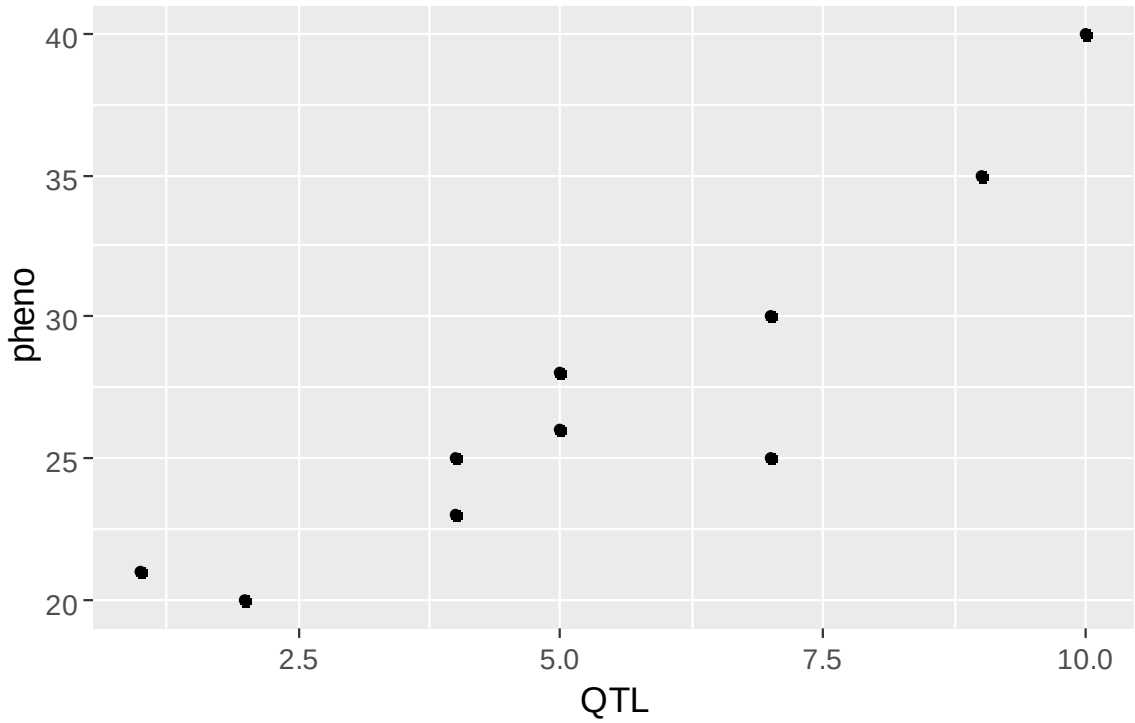
```
reg <- read.table("qtl.txt", header = T)
str(reg)
'data.frame':  10 obs. of  2 variables:
 $ QTL  : int  2 4 1 5 9 4 10 5 7 7
 $ pheno: int  20 23 21 28 35 25 40 26 30 25
```

Percentage of phenotype and number of quantitative trait loci in crop plants

Regression example

Plot first

```
ggplot(data = reg, aes(x = QTL,y = pheno)) +
  geom_point()
```



Regression: example

lm(y ~ x)
response ~ explanatory

```
> mod <- lm(data = reg, pheno ~ QTL)
> summary(mod)
```

Using the data argument makes it easier

Regression: example

```
> mod <- lm(data = reg, pheno ~ QTL)
> summary(mod)
Call:
lm(formula = pheno ~ QTL, data = reg)

Residuals:
   Min      1Q Median      3Q     Max
 -5.50  -0.50   0.00   1.25   3.50

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  16.5000     1.8307   9.013 1.83e-05 ***
QTL           2.0000     0.3026   6.609 0.000168 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.61 on 8 degrees of freedom
Multiple R-squared:  0.8452,      Adjusted R-squared:  0.8259
F-statistic: 43.68 on 1 and 8 DF,  p-value: 0.0001677
```

Summary information about residuals

b_0 and b_1

$$y = 2x + 16.5$$

Regression: example

```
Call:
lm(formula = pheno ~ QTL, data = reg)

Residuals:
   Min     1Q Median     3Q    Max
 -5.50  -0.50   0.00   1.25   3.50

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  16.5000     1.8307   9.013 1.83e-05 ***
QTL           2.0000     0.3026   6.609 0.000168 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.61 on 8 degrees of freedom
Multiple R-squared:  0.8452,      Adjusted R-squared:  0.825
F-statistic: 43.68 on 1 and 8 DF,  p-value: 0.0001677
```

Test: $b_0 = 0$
Often not impt

Test: $b_1 = 0$
Always of interest

Test of ‘model’
Same as $b_1 = 0$
in single
regression

Proportion of y
explained by x

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  16.5000     1.8307   9.013 1.83e-05 ***
QTL           2.0000     0.3026   6.609 0.000168 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.61 on 8 degrees of freedom
Multiple R-squared: 0.8452,     Adjusted R-squared: 0.8259
F-statistic: 43.68 on 1 and 8 DF,  p-value: 0.0001677
```

Reporting the results:

The number of QTLs explained a significant amount of the variation in percentage of phenotype (ANOVA: $F = 43.7$; $d.f. = 1,8$; $p = 0.00017$). The regression line is %phenotype = 2QTL +16.5

Give the statistical result

Give the line

Prediction

- Using coefficients For a single x

```
> intercept <- mod$coefficients[1]
> slope <- mod$coefficients[2]
> slope * 4 + intercept
    QTL
    24.5
```

When QTL = 4, % phenotype explained = 24.5

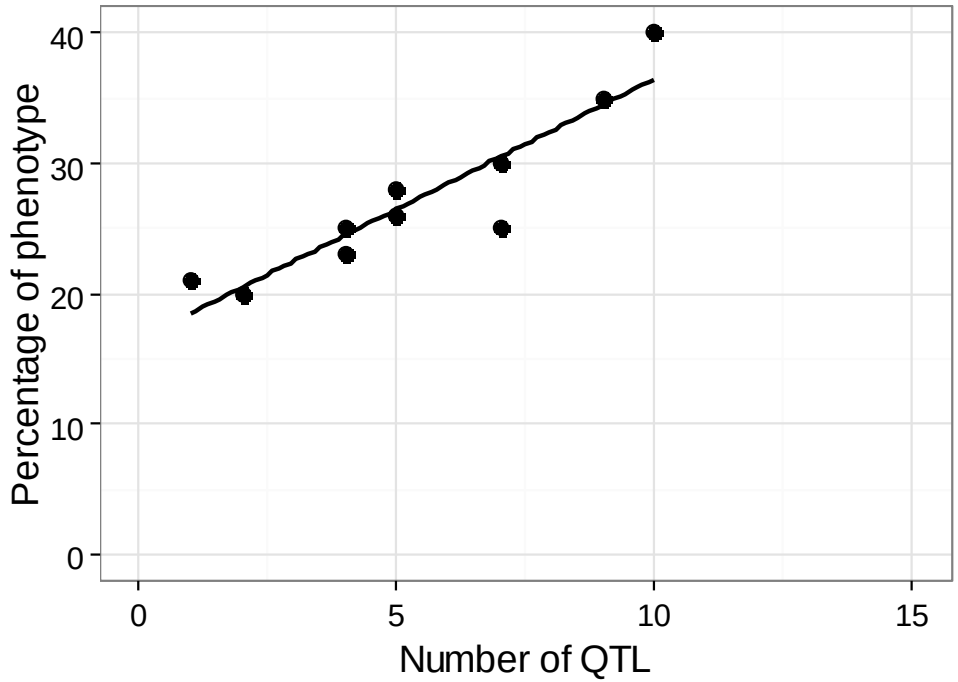
- More general –

```
> newdata <- data.frame(QTL = seq(2.5, 7.5,0.5))
> predict(mod, newdata)
 1    2    3    4    5    6    7    8    9   10   11
21.5 22.5 23.5 24.5 25.5 26.5 27.5 28.5 29.5 30.5 31.5
```

```
> newdata
   QTL
1  2.5
2  3.0
3  3.5
4  4.0
5  4.5
6  5.0
7  5.5
8  6.0
9  6.5
10 7.0
11 7.5
```

Regression figure

```
ggplot(data = reg, aes(x  =QTL, y = pheno))+
  geom_point(size = 2) +
  xlim(0, 15) +
  ylim(0, 40) +
  xlab("Number of QTL") +
  ylab("Percentage of phenotype") +
  geom_smooth(method = lm, se = FALSE, colour = "black")+
  theme_bw()
```



Assumptions of Regression

- Normality and homoscedascity of residuals
- *y* values are independent
- *x* is measured without error
- Linear regression assumes linear relationship

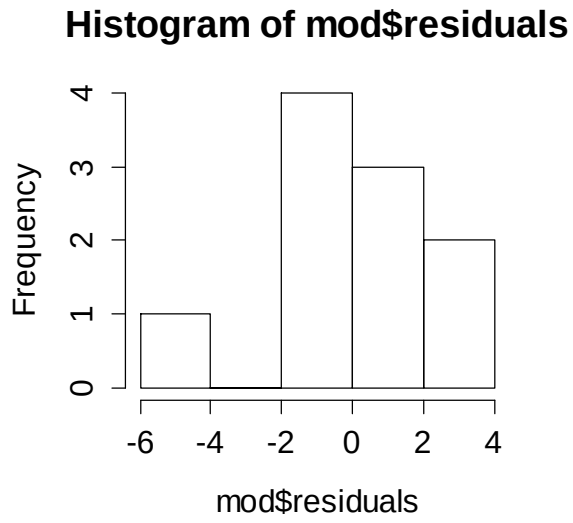
Testing the Assumptions Regression

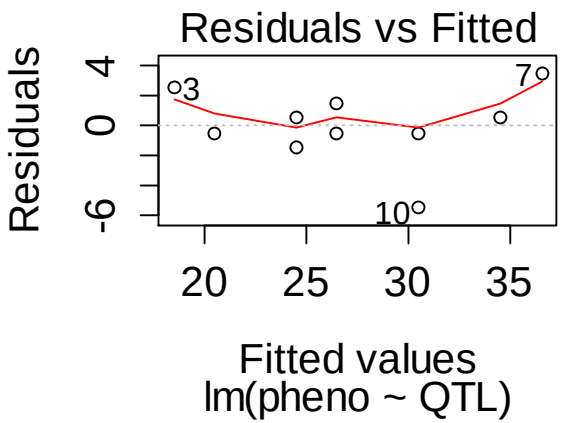
```
> hist(mod$residuals)
> shapiro.test(mod$residuals)

   Shapiro-Wilk normality test

data:  mod$residuals
W = 0.91739, p-value = 0.3357

> plot(mod, which = 1)
```



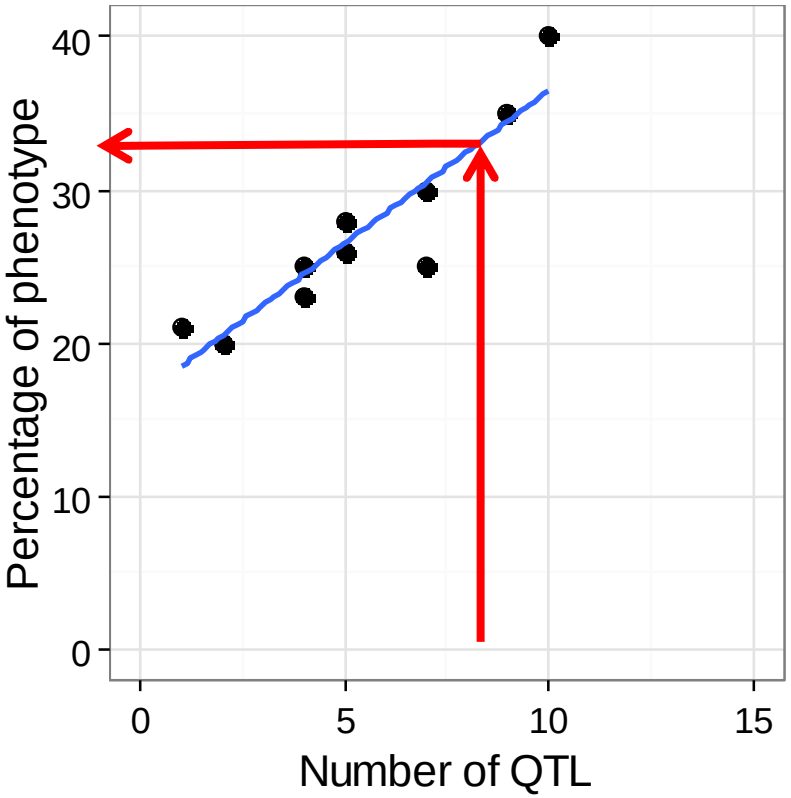


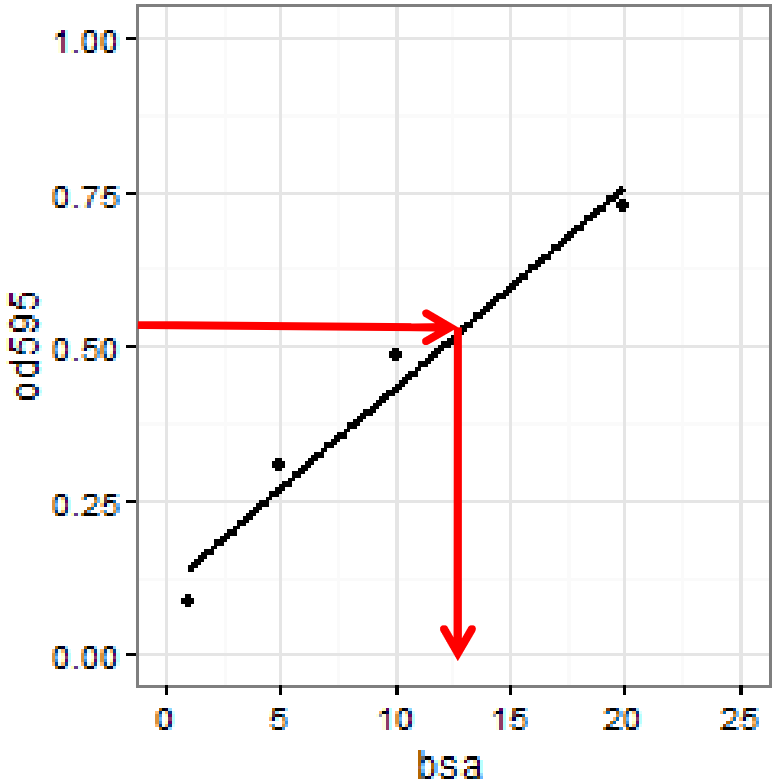
Small sample size but

these look OK

Linear regression for Calibration

- Regression
  - Set x measure y
  - Predict y for any value of x within range
- Calibration
  - Set x measure y
  - Predict unknown x from a measured y
  - AKA reverse /inverse regression
- Fundamentally the same


Percentage of phenotype
40
30
20
10
0
0
5
10
15
Number of QTL

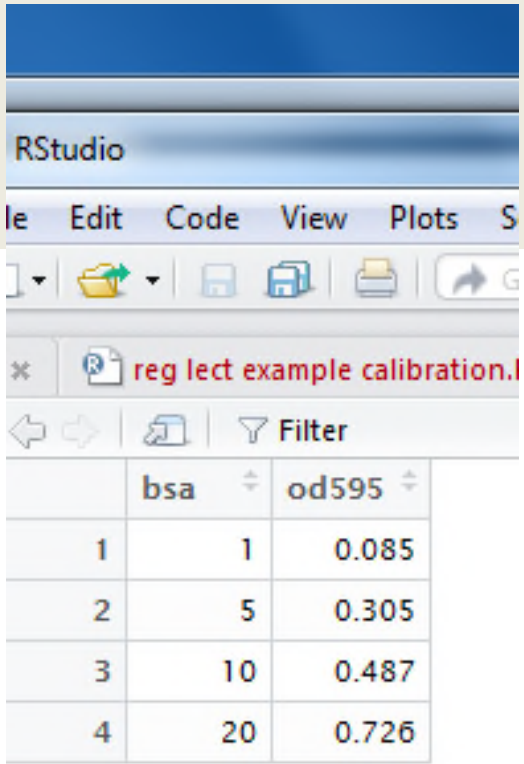

od595
1.00
0.75
0.50
0.25
0.00
0
5
10
15
20
25
bsa

Calibration example: estimating protein concentration

- Make a calibration 'curve'
  - known concentrations of Bovine Serum Albumin (BSA in µg) diluted with Bradford assay reagent
  - Measure optical density at 595nm
  - Perform regression Conc ~ OD
  - Line is OD= slope * Conc + intercept
- Predict unknown conc from OD:
  - Conc = (OD – intercept)/slope

Calibration example: estimating protein concentration

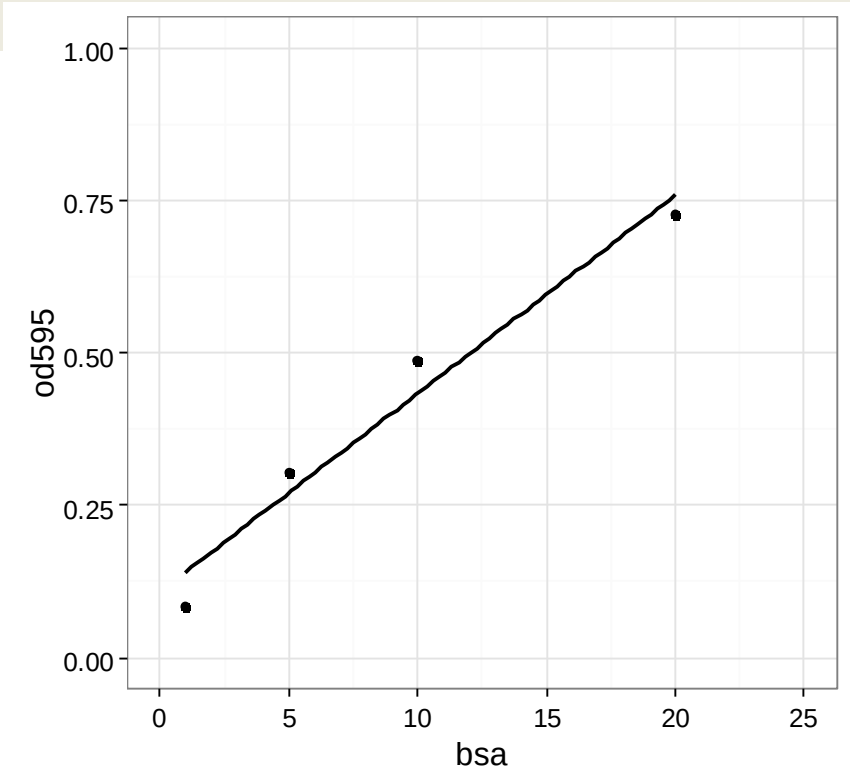
```
> standard <- read.table("../data/standard.txt", header=T)
> str(standard)'data.frame':  10 obs. of  2 variables:
'data.frame':  4 obs. of  2 variables:
 $ bsa  : int  1 5 10 20
 $ od595: num  0.085 0.305 0.487 0.726
```



	bsa	od595
1	1	0.085
2	5	0.305
3	10	0.487
4	20	0.726

Calibration ‘curve’

```
ggplot(data = standard,aes(x = bsa,y = od595)) +
     geom_point()+
     geom_smooth(method = lm,se = FALSE, colour = "black") +
     xlim(0,25) + ylim(0,1) +
     theme_bw(
```



Prediction

- Perform regression
- Access slope and intercept

```
calib <- lm(od595 ~ bsa, data = standard)
```

```
 (intercept <- calib$coef[1])
(Intercept)
  0.1078936
 (slope <- calib$coef[2])
   bsa
 0.0325396
```

- Evaluate for a particular OD
  - Suppose you measured absorbance at 0.4
  - Conc = (OD – intercept)/slope

```
 (0.4 - intercept) / slope
(Intercept)
  8.976951
```

Regression and Calibrations summary

- Regression - relationship
  - quote regression equation and test result (either ANOVA or *t*)
  - may also quote r^2
  - if scatterplot included do show a fitted line
- Calibration
  - Calibration curve
  - Give the predicted concentration