Summary toxicology

- Coding conventions
  - names are seperated with “.” (dotcase)
- Basic terminology
  - Definition of Toxicology
    - Study of adverse effects of chemical substances on living organisms.
    - Many touch on the relationship between doses and responses
  - Doses and responses
    - Every toxicological study involves a dosage of a compound and a measured response to the compound.
    - Example: gen expression against valporic acid
  - Alert concertration
    - What is it
      - The dosage at which something “interesting” happens.
    - How is it characterized
      - Percentages of maximum effect
      - Inference based approaches
        - Effects are significantly higher than prespecified response.
  - Studysetting
    - In Vivo
    - In Virto
    - In Situ
    - In Silico
- Isotonic regression (counterpart of antitonic regression)
  - Problem outline
    - Assuming monotonicity
    - How the optimal solution is characterized

$$\sum_{x \in X} w(x)(g(x) - f(x))^2 \quad \forall f \quad \text{isotonic}$$

▼ PAVA

▼ procedure

If for $j = i + 1 > i$ it holds, that $\mu(x_i) < \mu(x_j)$ then:

$$\mu^*(x_i, x_j) = \frac{n_i \mu(x_i) + n_j \mu(x_j)}{n_i + n_j}$$

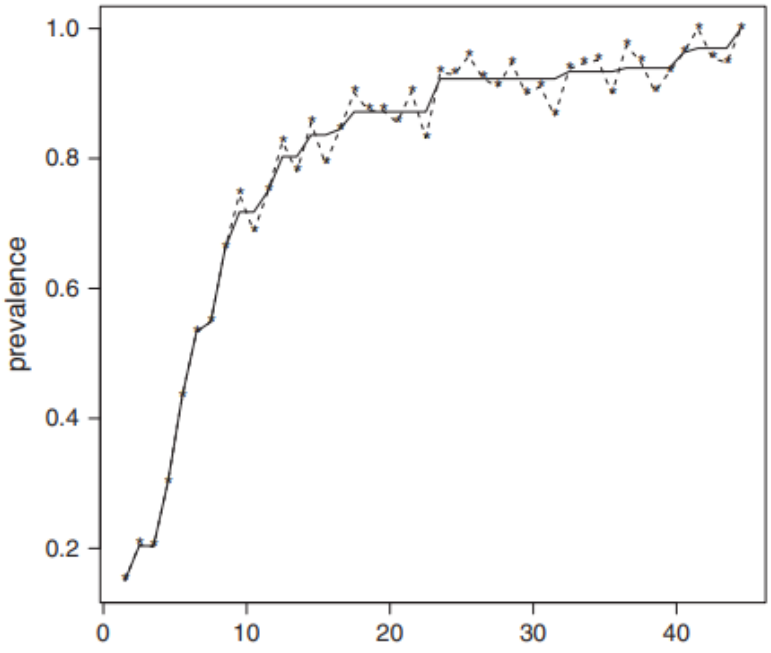
New estimates are replaced with $\mu^*(x_i, x_j)$

▼ example

Rubella → Dissease, you can only get once in your life

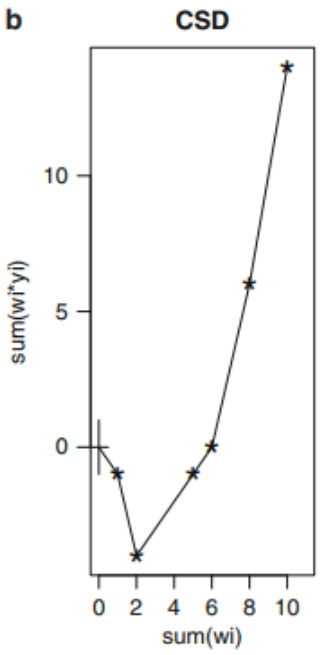
⇒ So it must follow, that prevelance after age must be monotonically increasing after age

applied results in:



▼ Cummulative sum diagramm (CSD)

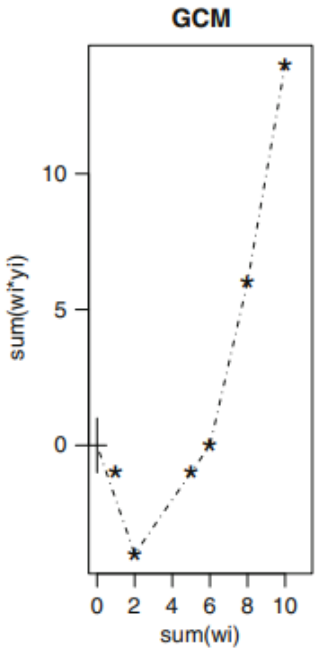
Let $W_i := \sum_{j=1}^{i} w_j$ where w_i are the weights of the means. Further more let $P_i := \sum_{j=1}^{i} w_j \mu(x_i)$ (sum over all weighted estimates). Then we define the CSD as all (W_i, P_i) for all $i \in \{1, ..., n\}$



If can be easily shown, that the slope between two points of this diagramm is equal to the estimate of the mean $\mu(x_i)$.

▼ greatest convex minorant (GCM)

Greatest convex function f, with: $f(W_i) \leq P_i \quad \forall i$



It holds that, estimates $\mu(x_i)$ get pooled iff $P_i > f(W_i)$

▼ Martix display

You can show, that $\mu^* = S\mu$ with S beeing a block diagonal matrix.

Example for a diagonal block martix:

$$\hat{\mu}^* = \boldsymbol{S}\hat{\mu} = \begin{pmatrix} 2/5 & 3/5 & 0 & 0 \\ 2/5 & 3/5 & 0 & 0 \\ 0 & 0 & 4/9 & 5/9 \\ 0 & 0 & 4/9 & 5/9 \end{pmatrix} \begin{pmatrix} \hat{\mu}_1 \\ \hat{\mu}_2 \\ \hat{\mu}_3 \\ \hat{\mu}_4 \end{pmatrix}$$

▼ Curve and model fitting

▼ continuous response data

▼ Sigmoid curves

Mathematical setting:

- The isotonic regression assumed, that $x_i$is ordinally scalled. Now we are going to assume that $x_i : \mathcal{A} \to \mathbb{R} \quad \forall i$ (so x_i is a random variable, that maps into the real numbers)

Why should you do this fit?

- Applying sigmoid curves is the first instance where we are trying to characterize the **alert-concentration.** In this setting the alert-concentration is defined by the mean of the upper and lower asymptote.

Things to consider when looking at research containing sigmoid fits:

- What parametrization was applied?
  - The interpretation depends on what formula was used.
  - In research, many different names refer to the same model (4pLL) with a different parametrization

▼ 4pLL (4 parameter log logistic regression)

This model is one of the most important models when estimating dose-responses.

We assume, the relation between the responses Y_{ij} and the dosages x_i can be described by:

$$Y_{ij} = \theta_1 + \frac{\theta_4 - \theta_1}{1 + e^{(\theta_2 - x_i)/\theta_3}} + \epsilon_{ij}$$

- You can reduce the model by assuming fixed values for the different parameters
- For instance, if the response is the relative amount of living cells, we can safely assume, that the upper asymptote is 1. Thus the regression in use is only a 3pLL.

Fit is done nummerical but not further discussed in the lecture.

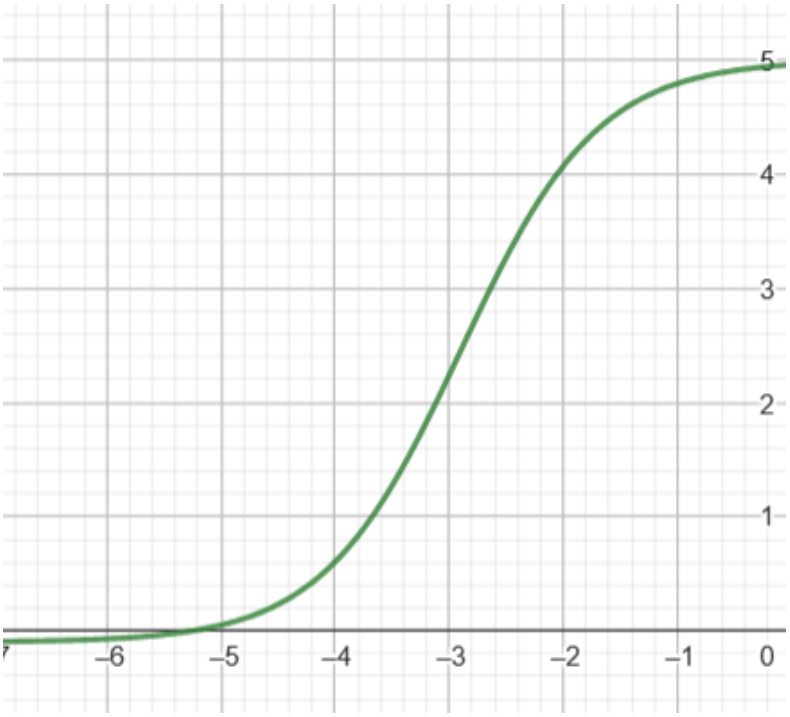
▼ different parametrizations

$$\begin{aligned} f(x, b, c, d, e) &= c + \frac{d - c}{1 + \exp(b(\log(x) - \log(e))} \\ &= c + \frac{d - c}{1 + \left(\frac{x}{e}\right)^b} \\ &= c + \frac{d - c}{1 + \exp(b(\log(x) - \tilde{e})} \end{aligned}$$

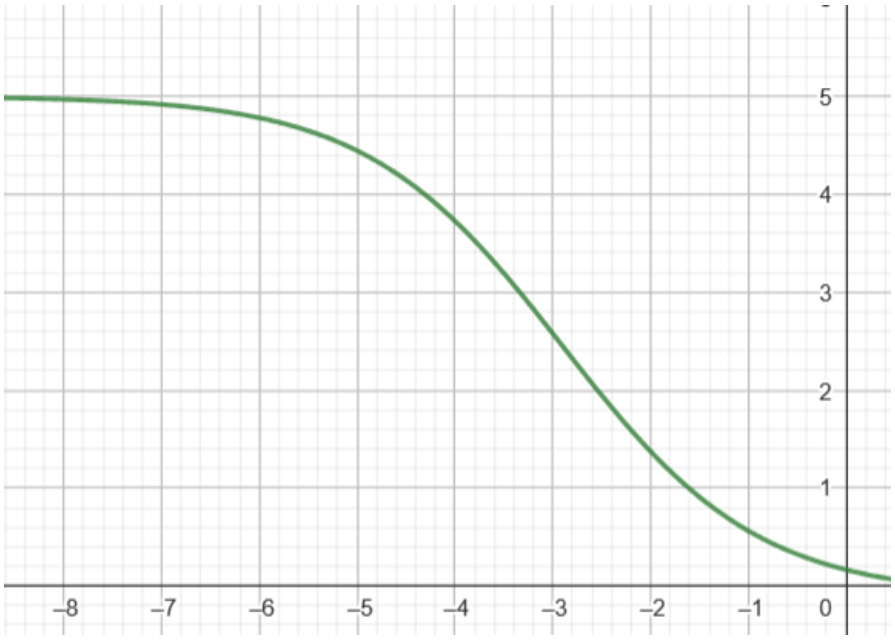
Where $\tilde{e} := \log(e)$

▼ Interpretation

- x_i **log scaled**
- θ_1, θ_4: lower and upper asymptote
- θ_3: slope of the function
  - **if the sign changes the roles of upper and lower asymptote reverse!!**
  - for $\theta_3 > 0$:



- for $\theta_3 < 0$:



- θ_2: is the dosis at which the effect reaches $\frac{|\theta_1 - \theta_4|}{2}$

▼ Application

Things to consider, when applying the model:

- **machine error:**
  - Because the fit is done numerically, we need to consider the machine epsilon and errors when calculating with floating point numbers.
  - **avoid very small values combined with very big values**
    - Possible solution: estimate $-\frac{1}{\theta_3}$ for the slope instead of θ_3

Applying the regression in R

- use the **gnls** function from the **nlme-package**

```
gnls.model001 <- gnls(ratio~(th1+(th4-th1)/(1+(exp((lmpk-th2)*th3)))),
                      data=data2b, params=list(th1+th2+th3+th4~1),
                      start=c(90,-0.2,1,28), control=gnlsControl(nlsTol=0.1))
```

Using drc package

```
model.fit <- drm(dose ~ response, data = data, curveid = group, fct=LL2.4())
```

▼ Other variants

▼ (sigmoidal) Emax Model

Another way of parametrization for the 4pLL is:

$$Y_{ij} = E_0 + \frac{x_i^n E_{max}}{x_i^n + ED50^n} + \epsilon_{ij}$$

- E_0: the baseline effect of the compound
- E_{max}: The maximum difference between the response and E_0 ($\theta_4 - \theta_1$)
- n: the slope parameter (can also be equal to $-\theta_3$)
- $ED50$: just like θ_2 in 4pLL

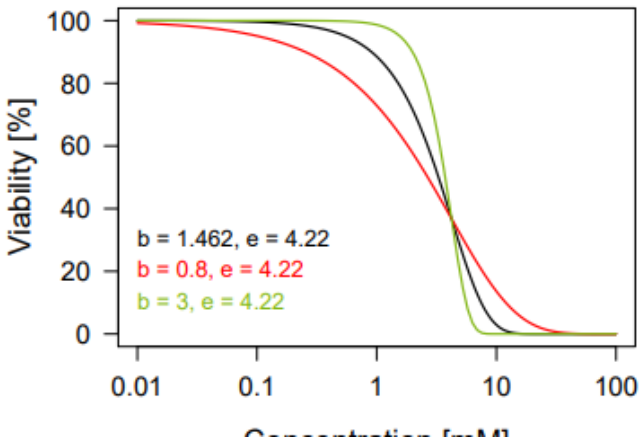
This model is a bit better to interpret

Another way of generalizing the 4pLL is to define a function f such that f : $\mathbb{R} \to \mathbb{R}$. Now the 4pLL can be written as:

$$f(x, a, b, c, d) = c + (d - c)f(b(log(x) - log(e))$$

- For $f = \phi$ we get the **log-normal regression**

- For $f = \exp(-\exp(x))$ we get a **weibull type 1 regression**
  - This regression has a slow descent at upper asymptote and a faster descent at the lower asymptote:



- for $f = 1 - \exp(\exp(x))$ we get a **weibull type 2 regression**
  - This regression has the oppsosite asymmetry compared to weibull type 1

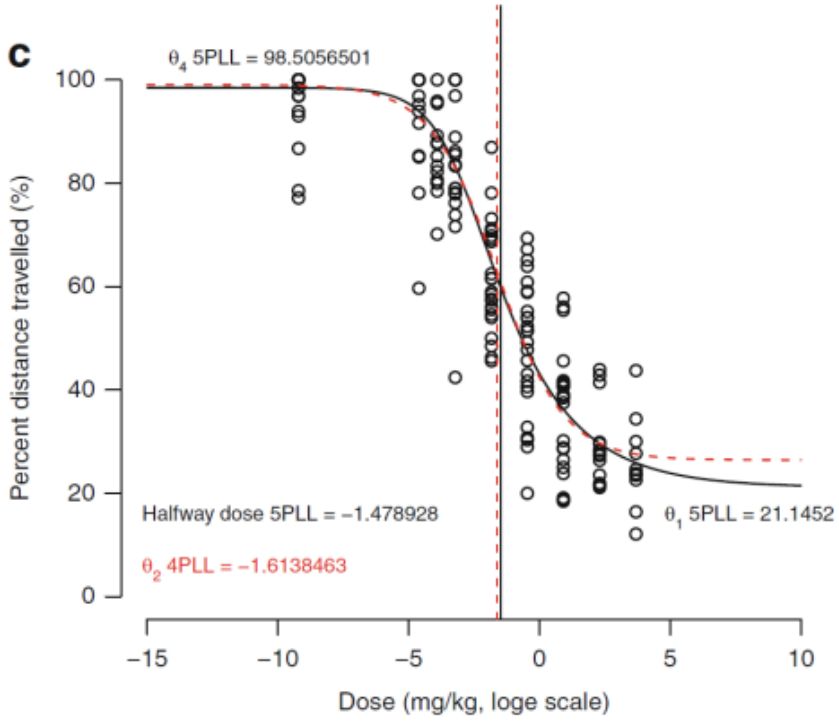
▼ 5pLL

▼ Controll over tails

New formular:

$$Y_{ij} = \theta_1 + \frac{\theta_4 - \theta_1}{(1 + e^{(\theta_2 - x_i)\theta_3})^{\theta_5}} + \epsilon_{ij}$$

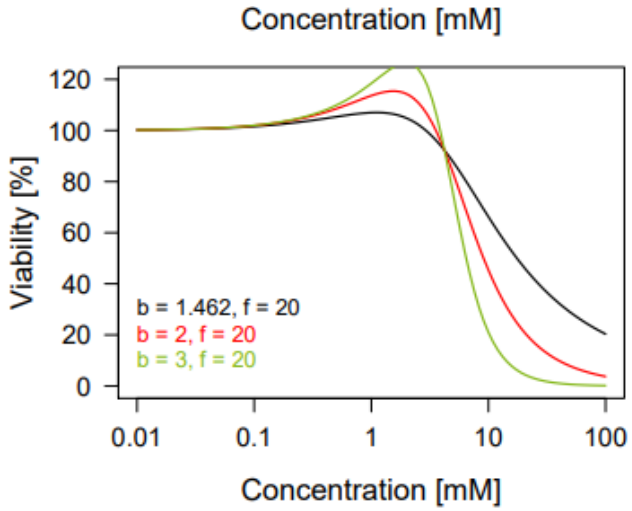
- With this model you sacrafice interpretability for more accuracy.
- You gain more controll over the tails of the sigmoid:



▼ Brain-Cousens-Model

$$f(x, b, c, d, e, f) = c + \frac{d - c + fx}{1 + \exp(b(\log(x) - \log(e))}$$

Models hormesis-effect, an initial positive effect, followed by a negative effect:



- f descirbes a the strength of the hormesis effect

- estimation
  - Covariance matrix
    - estimating under variance homogeniety

$$cov(\hat{\beta}) = \hat{\sigma}^2 \left(\left\{ \frac{\partial^2 f}{\partial \beta_{p_1} \partial \beta_{p_2}} \right\}_{p_1, p_2 \in \{1, \ldots, p\}} \right)^{-1}$$

      Is the scaled inverse of the information matrix.

      While the scale is determined by the variance of the residuals.

      We can use the unbiased variance estimator:

$$\hat{\sigma}^2 = \frac{1}{n-k} \sum (y_{ij} - f(x_{ij}, \hat{\beta}))^2$$

    - estimating under varaince heterogeniety
      - Sandwich estimator

        We denote the inverse of the information matrix:

$$\hat{B} := \left(\left\{ \frac{\partial^2 f}{\partial \beta_{p_1} \partial \beta_{p_2}} \right\}_{p_1, p_2 \in \{1, \ldots, p\}} \right)^{-1}$$

        And $\hat{M}$ the derivatives of the log likelyhood function

        Then:

$$cov(\hat{\beta}) = \hat{\sigma}^2 \hat{B} \hat{M} \hat{B}$$

      - Box-Cox transformation

        Transform data such that variance homogeniety is preserved.

      In both approaches we get lower estimates for the variance. Therefore p values are more signficant and confidence intervalls are smaller

  - confidence intervals

$$\hat{\beta}_i \pm K \hat{se}(\hat{\beta}_i)$$

Where K is the $1-\alpha/2$-quantile of the t-distribution, with degrees of freedom being the difference between number of observations and parameter count

▼ Model comparison

We will do a test if the models are equal. So given f_1 and f_2 we look at:

$$H_0 : f_1 = f_2 \quad H_1 : f_1 \neq f_2$$

▼ F-Test

$$F = \frac{\frac{RSS(f_1) - RSS(f_2)}{r-l}}{\hat{\sigma}^2_{f_2}} \overset{H_0}{\sim} F_{r-l;n-r}$$

▼ Implementation

Given two models `model.1` and `model.2` we can use the `anova(..)` function:

```
anova(model.1, model.2)
```

▼ Binary and binomial response data

▼ Setup

For given dosages $x_0 < x_1 < ... < x_k$ we have n_i different observations with a binary outcome w_{ij}. In general we can assume $W_{ij} \sim Bernoulli(p_i)$ So:

$$w_{ij} := \begin{cases} 1 \text{ if observed event occurs} \\ 0 \text{ else} \end{cases}$$

Then we estimate $p_i = P(W_{ij} = 1) = f(x_i, \beta)$

We can fit a variety of models, that fit the curve. If we do maximum likelyhood fitting, the log likelyhood function looks a lot more different than in the continuous case.

▼ Implementation

Just use `type = “binomial”` as the parameter in the `drm()` function

```
model.fit <- drm(response ~ dose, fct=LL4p(fixed=c(....)), data = data, type = "binomial")
```

▼ Count response-data

We assume a poisson distribution

▼ taking into account over dispertion

overdispertion happens, when the variability of the model does not represent the variability of the data.

In the case of counting data use a negative binomial distribution.

▼ drc package

▼ drm function

- documentation: drm function - RDocumentation
- constrainted estimation is possible but not advised (because you sacrafice model-performance)
- curveid and pmodels for fitting of multiple curves

▼ general setup

We assume:

$$Y_i = f(x_i, \beta) + \varepsilon_i$$

with $\varepsilon \sim \mathcal{N}(0, \sigma^2)$

- Note, that σ^2 has no index
- We assume the same variance over all x_i.

▼ model functions

Model type	Model function (*f*)	Function in *drc*
Generalized log-logistic	$c+\frac{d-c}{(1+\exp(b(\log(x)-\log(e))))^f}$	`llogistic()`
Brain-Cousens	$c+\frac{d-c+fx}{1+\exp(b(\log(x)-\log(e)))}$	
Cedergreen-Ritz-Streibig	$c+\frac{d-c+f\exp(-1/(x^a))}{1+\exp(b(\log(x)-\log(e)))}$	`cedergreen()`
(0 < α < 1 is usually fixed in advance)		
Log-logistic fractional polynomial	$c+\frac{d-c}{1+\exp(b(\log(x+1))^{p1}+e(\log(x+1))^{p2})}$	`fplogistic()`
Log-normal	$c+(d-c)\Phi(b(\log(x)-\log(e)))$	`lnormal()`
(Φ: distribution function for a normal distribution)		
Weibull I	$c+(d-c)\exp(-\exp(b(\log(x)-\log(e))))$	`weibull1()`
Weibull II	$c+(d-c)(1-\exp(-\exp(b(\log(x)-\log(e)))))$	`weibull2()`
Gamma	$c+(d-c)\bar{\Gamma}(bx,e,1)$	`gammadr()`
($\bar{\Gamma}$: distribution function for a Γ distribution)		
Multistage	$c+(d-c)\exp(-b_1-b_2x-b_3x^2)$	`multi2()`
NEC	$c+(d-c)\exp(b(x-e))$ for $x>e$ and d otherwise	`NEC.4()`

- we can give fixed values with the “fixed” parameter
- models are well defined for $\lim_{x\to-\infty} f(x)$ (because doses are log scaled and we need a response value for the negative-dose (dose = 0))

▼ LL.4()

$$f(x,b,c,d,e)=c+\frac{d-c}{1+\exp(b(\log(x)-\log(e)))}$$

▼ LL.5()

$$f(x,b,c,d,e,f)=c+\frac{d-c}{(1+\exp(b(\log(x)-\log(e))))^f}$$

▼

▼ Estimation

▼ parameters

- Normaly maximum likelyhood is used
- Another way could be to look at the following non linear least squares problem:

$$\sum_{i=1}^{n} w_i^2(y_i-f(x_i,\beta))^2$$

- With $f(x_i, \beta)$ being the used model $f : \mathbb{R}^{k+1} \to \mathbb{R}$ with parameters $\beta \in \mathbb{R}^k$
- we can use the **gauß-newton-algorithm** to find te β that minimizes this problem
- However it is only required to know that this is a **numerical procedure** and we need starting values.

▼ covariance matrix

- We will calculate the covariance for the estimates with the second partials of model f:

$$\operatorname{cov}(\hat{\beta}) = \hat{\sigma}^2 \left(\left\{ \frac{\partial^2 f}{\partial \beta_{p_1} \partial \beta_{p_2}} \right\}_{p_1, p_2 \in \{1, \dots, p\}} \right)^{-1} = \hat{\sigma}^2 \left(\frac{\partial^2 f}{\partial \beta_{p_1} \partial \beta_{p_2}} \right)^{-1}$$

Whereas the scaling is $\hat{\sigma}^2 = \frac{1}{n-p} \sum_{i=1}^{n} (y_i - \hat{f}(x_i, \hat{\beta}))^2$ (residual variance, adjusted to be unbiased)

Estimating with log likelyhood

We can also take the inverse of the log-likelyhood function:

$$cov(\hat{\beta}) = \left(\frac{\partial^2 l}{\partial \beta_{p_1} \partial \beta_{p_2}} \right)^{-1}$$

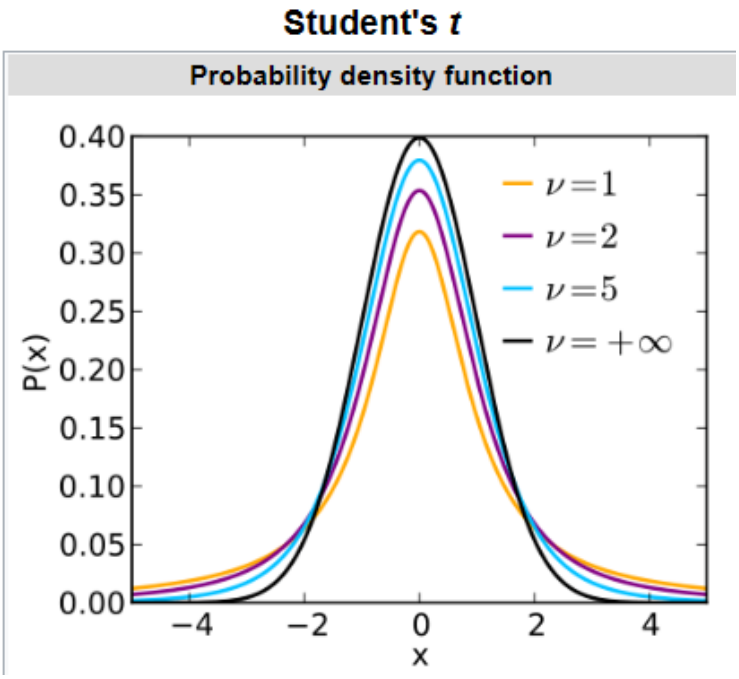
▼ Confidence intervals

Intervals for the estimates are calculated by:

$$\hat{\beta}_i \pm K \hat{se}(\hat{\beta}_i)$$

whereas K is the **quantile of the response datas distributions**

- In the case of normally distributed response data, we assume K to be a $(1 - \alpha)$ -Quantile of a t distribution



    - t-distribution is just a generalized normal distribution
- If we assume $K = \phi_{1-\alpha}$, a quantile of the standard normal distribution.

▼ ED values (effective dose)

Inverse value for specific percentage α of maximum effective dosage $ED100$:

$$f(ED100\alpha, \beta) = (1-\alpha)\lim_{x\to 0} f(x,\beta) + \alpha \lim_{x\to\infty} f(x,\beta)$$

- There is a differenciation between increasing and decreasing curves

However sometimes, it is not possible to reach the $ED100$ dose (concentration can't be higher under fixed temperature).

In these cases we can use **absolute** effective dosages:

$$f(ED100_{100\lambda}, \beta) = \lambda f(ED100_{100\lambda}, \beta)100\lambda$$

Comparing ED-values:

▼ Sandwich estimator

▼ Box Cox

Original model is shifted by constant C and transformed by real function $g_\lambda(x)$

- Box Cox is for variance heterogenity
- The difference to the other methods is, that we transform the r**elationship** between the explanetory variable and endogenic variable

▼ assumptions for distribution

scaling:

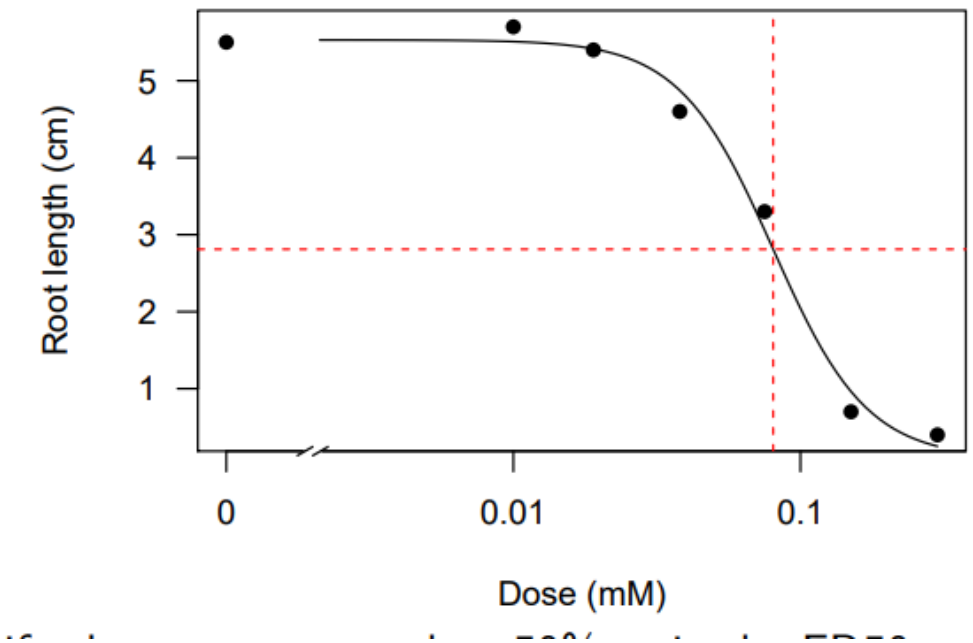
- continuous
- discrete
  - binomial
  - binary
- counting
  - Poisson distribution assumed

▼ examples

▼ continuous response data (plant growth inhibition)

- $Y_i = f(x_i, \beta) + \varepsilon_i$, for $i = 1, \ldots, n$, where $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$

- Normal destribution with residuals iid and homogenious variance
- **data**
  - effect of compound **(secalonic acid)** on **plant growth**
  - plant growth in root length
- Model after fit:




```
>
> summary(secalonic.LL.4)

Model fitted: Log-logistic (ED50 as parameter) (4 parms)

Parameter estimates:

                Estimate Std. Error t-value   p-value
b:(Intercept) 2.6542086  0.6962333  3.8122 0.0317398 *
c:(Intercept) 0.0917852  0.3747246  0.2449 0.8223012
d:(Intercept) 5.5297495  0.2010300 27.5071 0.0001055 ***
e:(Intercept) 0.0803547  0.0078829 10.1935 0.0020121 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error:

 0.2957497 (3 degrees of freedom)
>
```

- We should not reduce to a 3pLL model, because such a decision has to be done beforehand.
    - Do not base your decisions on previous modelfits

▼ continuous response data (growth of ryegrass)

▼ counting response data (flea reproduction)

Some compound on the reproduction of fleas

Time is used as weights

- ▼ Things to take into account
  - ▼ Over dispersion

    Variaty of data does is not captured in the model

    - We encountered this problem with counting data
    - one solution was to use a negative binomial distribution instead of Poisson
      - Using this will result in higher std. errors. But this is just a sign, that we now capture the full variance
  - ▼ under dispersion

    Variaty of data is lower than the models variaty

- ▼ Estimating Alert concentration
  - ▼ ED-Values
    - ▼ Relative

      Notation $ED100\alpha$. Is the dosage at which α of the maximum effect is reached:

      $$f(ED100\alpha) = \lim_{x \to 0} f(x, \beta) + \alpha \left(\lim_{x \to \infty} f(x, \beta) - \lim_{x \to 0} f(x, \beta) \right)$$

    - ▼ Absolute

      Is a dosage at which a fixed effect λ or 100λ is reaced. Notation $ED_{100\lambda}$.

      $$f(ED_{100\lambda}, \beta) = \lambda \text{ or } f(ED_{100\lambda}, \beta) = 100\lambda$$

    - ▼ Relative potency

      Is the ration between two effective dosages for two different curves $ED_A100\alpha$ and $ED_B100\alpha$.

      $$\text{realtive potency} = \rho(\alpha, \alpha) := \frac{ED_A100\alpha}{ED_B100\alpha}$$

  - ▼ Benchmark Dose Estimation (BMD)

    Find the lowest dose at which there is a deviation from the normal response

    - ▼ Background level / risk

      Can be estimated directly from the data or from literature

    - ▼ Benchmark response risk (BMR)

- Usually 10%

▼ Estimating for binomial response data

- Easier to estimate because 10% level always known
- excess or extra risk is recommended

$$BMR = \frac{f(x, \beta) - p_0}{1 - p_0}$$

  - Percentage of difference between BR and maximum value

▼ Estimating for continuous data

Dichotom approach

We can define a cutoff x_0 where for any repsonse y_i we can define:

$$v_i = \begin{cases} 1 \text{ if } y_i > x_0 \\ 0 \text{ else} \end{cases}$$

Hybrid approach

Let x_0 be a fixed value on response scale that is an abnormal response. Furthermore we will say, that $Y \sim \mathcal{N}(f(0, \beta), \sigma^2)$ and $Y\text{‘} \sim \mathcal{N}(f(BMD, \beta), \sigma^2)$. Then we can say:

$$BMR = \frac{P(Y\text{‘} \geq x_0) - P(Y \geq x_0)}{1 - P(Y \geq x_0)}$$

For an unknown x_0 we use $2\sigma^2$ by convention

▼ ALEC (Absolute lowest effective concentration)

Absolute lowest effective concentration $(ALEC)$ for value λ. So:

$$f(ALEC, \beta) = \lambda \implies ALEC = f^{-1}(\lambda, \beta)$$

Because if f is invertible, we have a function $h(\lambda) = f^{-1}(\lambda)$ for calculating the $ALEC$.

▼ Estimation under delta method

Under the assumption $X \sim \mathcal{N}(\mu, \sigma)$ it follows that $f(X) \sim \mathcal{N}(f(\mu), \sigma^2 f'(\mu)^2)$. We get this result by first order taylor approximation. Especially $Var(f(X))$ is interesting and we can expand this result to a multidimensional case:

$$X \to \mathbb{R}^d \implies Var(f(X)) = \nabla f(X)^T \cdot \Sigma \cdot \nabla g(\beta)$$

From this we get the estimation for the $1-\alpha$ confidence intervall:

$$\widehat{ALEC} \pm t_{\nu,(1-\alpha/2)}\sqrt{\widehat{var}(\widehat{ALEC})}$$

While $t_{\nu,(1-\alpha/2)}$ is the $1-\alpha/2$ quantile of the t-distribution with $n-k$ degrees of freedom (n being the sample size and k being the number of parameters).

▼ ALOEC and LOEC

With these alert-concentrations we only consider actual measured dosages.

ALOEC → first time, where the mean exceeds threshhold of $f_0 + \lambda$

LOEC → first dosage, where the response is **significantly** higher than $f_0 + \lambda$

- More on this topic in the testing part of stat in tox.

▼ LEC

Lowest concentration x where $f(x, \phi)$ is **signficantly larger** than $f_0 + \lambda$.

- In the Example we used the t-test due to $\hat{\phi}$ being (by assumption) normally distributed

▼ Comparison of observation based and model based alert concentration

▼ Examples

▼ Pathogens in food

Exposure to virus

Response data: percentage of infections

Therefore: 2pLL model

▼ Chromosomal damage

Potential damage to chromosomes after exposure to compound.

Response: number of cells with damage

Because damage can occur without exposure, we need to estimate background risk

Also damage does not reach all cells

Therefore: 4pLL

BMR: 1%

▼ implementation

▼ Bmd package

```
install_github('DoseResponse/bmd')
```

▼ Optimal Design Considerations

▼ What is an optimal design

An optimal design gives the most information for curve fitting. Each observation maximizes it's leverage on the parameter estimation, meaning that the gradient for the observation gets maximized.

Problem:

The optimization-problem requires us to maximize a matrix. But there are several ways of maximizing a matrix:

- A-optimal: The sum of it's diagonal enteries meaning the trace of the matrix
- C-optimal: Maximize variance for one specific point
- D-optimal: Maximize the determinant of the matrix

For D-optimal there is a way of characterizing the optimal solution. Let $x_1, ..., x_m$ be the possible dosages. Furthermore let a design where every obsevation measures the same dosage be defined as an **elementary design** in this case the information-matrix is denoted $A_j \forall j \in \{1, .., m\}$**.** The optimal design yields an information matrix M_o that:

$$\varphi_j := \text{trace}(A_j M_o^{-1})/4 \leq 1 \quad \forall j \in \{1, ..., m\}$$

▼ Algorithmic approach for D-optimal designs

Start with dose-levels $x_1, ..., x_m$ and some primitive weights $w_{11}, ..., w_{m1}$. Then let's denote for the first iteration φ_1. We will repeat:

- compute φ_{jn} for all j. And compute: $w_{jn} = \varphi_{jn} w_{j,n-1}$
- if $\frac{\varphi_{jn}}{\varphi_{j,n-1}} < \lambda \forall j$ stop

▼ D-Efficiency

We can denote the efficiency of an experimental design by:

$$E_{design} = \sqrt[k]{\frac{\det(M_{design})}{\det(M_{opt})}}$$

Interpretation of E_{design}:

- You need E_{design}^{-1} many samples as the optimal design to get the same percision of the modelfit.

▼ Model Selection

▼ MCP-Mod (Multiple comparison procedures and modelling)

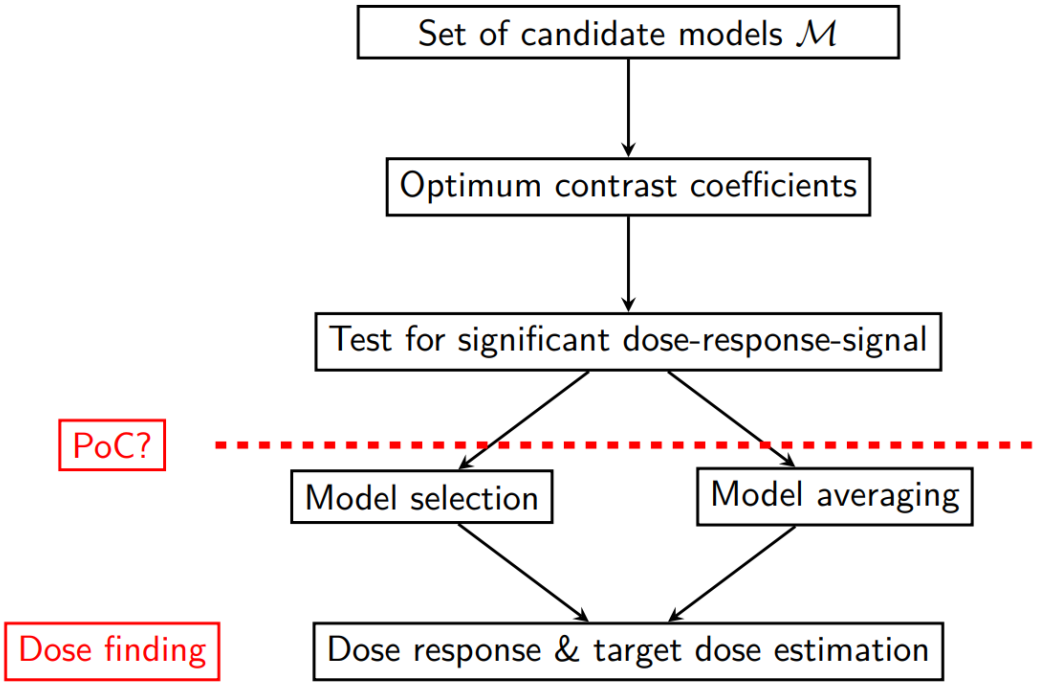
▼ Components:

- Only fit model if a signal is present (dose response curve is not flat) (MCP)
- If signal is present fit multiple models and estimate relevant dosages (Mod)

▼ Weaknesses of components

- If we fit a model to every response-curve, flat curves might bias the minimum dosage upwards
- If we only use a one way ANOVA to check for a flat curve, we might incorporate noisy data into the final estimates.

▼ Combination:



- Modelling pre-specified to design stage

▼ Perform for set of candidate models $\mathcal{M}$

- $Y_{ij} = \mu_{d_i} + \epsilon_{ij} = f(d_i, \theta) + \epsilon_{ij}$
- Here $f(d_i, \theta)$ is the model
- We will use a **guestimate approach**
  - This means, the toxicologist has some rough idea, of what should happen at what dosage
  - The Model then can be written as

$$f(d, \theta) = \theta_0 + \theta_1 f^0(d, \theta^0)$$

  - While f^0 is all non linear parts of the model
  - Now the toxicologist may say, that dose d_1, d_2 will lead to effects (as percentage of maximal effect) p_1, p_2 and therefore we convert this model to:

$$p_1 = f^0(d_1, \theta^0) \quad (1)$$
$$p_2 = f^0(d_2, \theta^0) \quad (2)$$

  - As we can see, the θ_0 is cancelled out and can use the equations to solve for θ^0.

▼ Compute optimal coefficients

- We will choose m **contrast-coefficients** for the $M = |\mathcal{M}|$ models which we computed.
- Let c_{mi} with $i \in \{1, ..., k\}$ be the contrast coefficient for the mth model and the ith dosage
- We use these coefficients, to optimize the power of the test-tatstic, which will detect, whether the dose-response-curve is flat or not.
- Under optimization we can say:

$$c_{mi} \propto n_i(\mu_{mi}^0 - \bar{\mu}_m)$$

▼ Test for signal afterwards (MCP) / Proof of concept step

- We will use the contrast test, computed in the step from before to test for flat curves

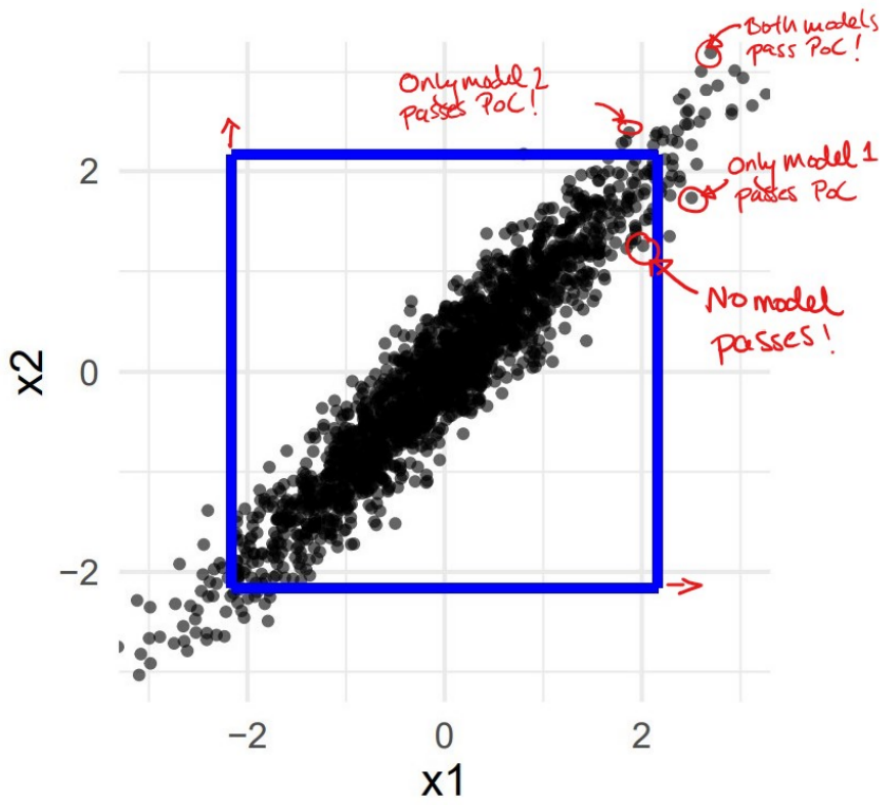
- The final test-statistic is:

$$T_{max} = \max\{|T_1|, ..., |T_M|\}$$

- We then have proof of concept as soon as:

$$T_{max} \geq q_{1-\alpha/2}$$

- $q_{1-\alpha/2}$ is computed by sampling from the test-statistics $T_1, ..., T_M$ many times and choosing $\hat{q}_{1-\alpha/2}$ such that $1-\alpha$ percent of the observations are within $[0, \hat{q}_{1-\alpha/2})$.
- The test-statistic $T = (T_1, ..., T_M)'$ follows a multivariate t-distribution. $\hat{q}_{1-\alpha/2}$ is then a equicoordinate quantile of this distribution. Here we can see it for a 2-dimensional distribution:



▼ Model selection

- We could now choose the model m with the highest test-statistic ($T_m = T_{max}$).

- …or we could the AIC and BIC values of the models
  - AIC rewards goodness of fit (more popular)
  - BIC rewards goodness of fit but takes into account sample size
- If proof of concept is present and model is chosen, fit model

▼ Dose estimation

▼ Model selection approach

- use model according section above
- Estimate MED (minimum effective dosage) by taking into account the confidance interval. We will denote the lower bound of the confidence interval for a given dose d: $L(d)$:

$$\widehat{MED} = \min\{d \in (d_1, d_k] | \hat{f}(d) > \hat{f}(d_1) + \Delta \wedge L(d) > \hat{f}(d_1)\}$$

- Where Δ is some clinical threshold
- This estimation is safer towards errors introduced by the fitted model

▼ Model averaging

We will use the weighted average over all models **with proof of concept.** Let $\mathcal{M}^*$ be the models with proof of concept and $k = |\mathcal{M}^*|$:

$$\widehat{MED} = \sum_{i=1}^{k} w_i \widehat{MED}_i$$

The weights are chosen to their performance in AIC or BIC respectively.

$$w_I = \frac{p_I \exp(-IC_I \cdot 0.05)}{\sum_{j=1}^{L} p_j \exp(-IC_j \cdot 0.05)}$$

This the same formular we used in the section **Model-averaging.**

▼ Implementation

The following implementation follows the MCPMod package.

```
install.packages("MCPMod")
```

▼ Guestimates

The is a function specifically is called `guesst(....)` :

```
# general usage:
guesst(d, p, model = c("emax", "exponential", "logistic", "quadratic",
"betaMod", "sigEmax"), less = TRUE, local = FALSE,
dMax, Maxd, scal)

# specific example
guesst(d = 700, p = 0.5, model="exponential", Maxd=1000)
```

- `d:` vector containing dose values
- `p:` Vector containing corresponding percentages of maximal effect or dose-response
- `model:` (vector) of strings, of which models to use
- `dMax:` maximal effect
- `Maxd:` maximal dosage
- `scal:` scale parameters (only for beta-model)

▼ Contrast statistic

The function `guesst(…)` will return the parameters to use. Afterwards we will use `mods(…)` and `optContr(….)` to compute the optimal contrast coefficients:

```
# using parameters from guesst
candMods <- Mods(betaMod = c(..), emax= c(...),....)
weights <- c(...)
contMat <- optContr(candMods, w = weights)
```

Here we however use functions, which are part of the **doseFinding** package available on CRAN:

```
install.packages("doseFinding")
```

▼ Testing

For this we use MCTtest(…), a function from the doseFinding package:

```
MCTtest(dose,
  resp,
  data = NULL,
```

```
  models,
  S = NULL,
  type = c("normal", "general"),
  addCovars = ~1,
  placAdj = FALSE,
  alpha = 0.025,
  df = NULL,
  critV = NULL,
  pVal = TRUE,
  alternative = c("one.sided", "two.sided"),
  na.action = na.fail,
  mvtcontrol = mvtnorm.control(),
  contMat = NULL)
```

There are a lot of parameters here we will only cover the most important:

- `dose, resp:` either names of `data.frame` of `data` parameter or actual dose response combinations (if we look at only one profile)
- `data: data.frame` with dose response data
- `models:` objects of class `Mods`
- `alpha:` level of significance

▼ Model selection

Here we make use of the `MCPMod(…)` function. For this we need to supply the models as `Mods` objects:

```
res2 <- MCPMod(dose,
  response,
  data,
  models = candMods,
  alpha = 0.05,
  alternative = "two.sided")
```

Usage is the same as in `MCTtest`.

▼ Model averaging

▼ AIC based weights

We want to use the probability that model f_r is the true model as the weight. Therfore we need to find a way to compute $P(f_r|D)$.

We first define the information criterion as a combination of the log likelyhood and a penalty-term for model-complexity:

$$AIC_r := -2 \log \mathcal{L}_r(\theta, D) + 2M_r$$

Where M_r is the number of parameters in the model. Then we denote ΔAIC_r the difference between the AIC value of model r and the minimum AIC value amongst the models:

$$\Delta AIC_r = AIC_r - AIC_{min} \tag{3}$$

By approximating $-2\mathbb{E}(\log(P(D|f_r)) \approx -2\mathbb{E}(\log(\mathcal{L}_r(\theta|D))) + 2M_r$ and using (1) we can say:

$$P(D|f_r) \propto \exp\left(-\frac{1}{2}\Delta AIC_r\right) \tag{4}$$

By using Bayes Theorem, we can now compute $P(f_r|D)$ which is now the probability, that f_r is the true model.

$$w_r = P(f_r|D) = \frac{P(D|f_r)P(f_r)}{\sum_{i=1}^{R} P(D|f_i)P(f_i)}$$

And because of (2) the constants cancel each other out and we can use (2) for $P(D|f_i) \forall i$.

Now we can use:

$$\hat{f}_{MA} := \sum_{i=1}^{R} w_r \hat{f}_r$$

▼ Clustering

Especially when working with gene-expression data, we want to cluster genes with similar dose-response-profiles.

Problem:

- KNN clusters curves that lie near each other but does not put as much weight on similar slopes.
- Solution: Biclustering

▼ Biclustering

Let Data-matrix $Y \in \mathbb{R}^{n \times (k+1)}$.

We model the data by assuming each gene has the same slope:

$$y_{mi} = \mu + \alpha_m + \beta_i + \epsilon_{mi}$$

- y = dose-response for mth gene and ith dosage (with controll)
- α_m fixed effect for dosage
- β_i fixed effect for gene

Then let H_Y be the MSE of the model.

The higher H_Y the more heterogeniety there is in the data-matrix

Now lets find $\tilde{Y}$ a submatrix of Y such that $H_{\tilde{Y}} < \delta$.

This is the rough idea. Most of the implementations rely on computing the contributions of each row and column and removing the ones with the highest.

For clustering of gene expression, we can allow to only remove full genes and no dosages.

Furthermore we need to have a minimum of genes per cluster (because removing all rows results in $H_{\tilde{Y}} = 0$.

Choosing a good δ

Usually it is hard to predict a good δ for this algorithm. It is recommended to let it terminate, when $H_{\tilde{Y}}$ behaved convergent with each iteration. So:

$$\frac{H_{\tilde{Y}}}{H_Y} \leq \lambda \quad 0 \leq \lambda \leq 1$$

Choosing a good number of clusters

Both the number of clusters n and the squared residuals W are dependent on δ or λ.

So the optimal λ minimizes:

$$\arg \min_{0 \leq \lambda \leq 1} W(\lambda) + n(\lambda)$$

▼ ORIOGEN

So far we have looked at clustering methods for monotone curves. This method is especially dedicated for identifying umbrella profiles (unimodal). ORIGEN → Order restricted inference for ordered gene expresion

1. define orders onto mean values
2. apply restricted estimation for each profile

3. compute test statistic for every profile
4. Approximate distribution (max of every profile) using bootstrap
5. if larger than some quantile, then assign profile with largest test statistic

- Also contains graph theory
- No implementation in R
- Assumption: residuals do not correlate
-

▼ ORICC

Use restricted estimation from profiles as model and evaluate information criterion.

ORIC → order restricted information criterion clustering

Uses **normal distribution and homogeniety among variance.**

let $\mathcal{L}(\hat{\mu}, \hat{\sigma}^2)$ be the log likelyhood function of the normal distribution for estimates $\hat{\mu}$. Then $ORIC(C)$ of order restriction C is defined as:

$$ORIC(C) = -2\mathcal{L}(\hat{\mu}, \hat{\sigma}^2) + p(C) \log\left(\sum m_i\right)$$

- $p(C)$ is a penalty-term for model complexity

$$p(C) = \nu_1 + \sum_{i=0}^{\nu_2} \frac{1}{1+i}$$

  - ν_1 number of $\neq$ inequalities (grows linear in ν_1, because we do not restrict
  - ν_2 number of $\{\leq, \geq\}$ inequalities (grows less than one with number of inequalities, because we restrict)
- m_i is the sample-size for each dose.

If $\hat{C} := \min_C ORIC(C) \neq C_0$ then choose $\hat{C}$

Two stage approach

Before estimation of all profiles and evaluation of ORIC, compare unrestricted profile with C_0. If $ORIC(C_0) < ORIC(C_{\neq})$ do not continue.

Has benefits for runtime.