Maximum Likelihood in R

The Basics

Estimation. `optim` is the engine that searches for parameters that minimize (or maximize) a function. Usually easier to access it via a convenience function like `mle2`.

Confidence. `profile` returns a likelihood profile, originating at a solution returned from `optim` (or a helper function). `confint` uses a profile (or computes a new one) in order to give profile likelihood confidence intervals. Can get quadratic approximate confidence intervals (estimated standard errors) by `sqrt(diag(vcov(m)))` (if you used `m <- mle2` for estimation) or `sqrt(diag(solve(o$hessian)))` (if you used `o <- optim` directly). Once you have these standard errors, multiply by the appropriate z-score (1.96 for 95% intervals, for example) to get size of deviation in either direction.

Plotting. `plot(profile(m))` will plot likelihood profiles for solutions in `m`.

Likelihood Examples

Starting parameter values in these examples are not general to all problems. As always, think about your data to find reasonable starting values.

Gaussian regression (OLS). $y_i \sim \mathcal{N}(\alpha + \beta x_i, \sigma)$

```
m <- mle2( y ~ dnorm( mean=a+b*x , sd=s ) ,
        start=list( a=mean(y) , b=0 , s=sd(y) ) )
```

Binomial model with intercept only. $y_i \sim \text{Binom}(p, N)$

```
m <- mle2( y ~ dbinom( prob=p , size=n ) , start=list(p=0.5) )
```

Binomial model with covariate (logistic transform). $y_i \sim \text{Binom}\left(\frac{\exp(\alpha+\beta x_i)}{1+\exp(\alpha+\beta x_i)}, N\right)$

```
m <- mle2( y ~ dbinom( prob=exp( a + b * x )/(1 +exp( a + b * x )) ,
     size=n ) , start=list(a=0,b=0) )
```

Poisson model with intercept only. $y_i \sim \text{Pois}(\lambda)$

```
m <- mle2( y ~ dpois( lambda=p  ) , start=list(p=mean(y)) )
```

Poisson model with covariate. $y_i \sim \text{Pois}(\alpha + \beta x_i)$

```
m <- mle2( y ~ dpois( lambda=a + b * x  ) , start=list(a=mean(y),b=0) )
```

Negative-binomal model. $y_i \sim \text{Nbinom}(m, n)$

```
m <- mle2( y ~ dnbinom( mu=m , size=n  ) , start=list(mu=mean(y),n=1) )
```

Beta-binomial model. $y_i \sim \text{BetaBinom}(\alpha, \beta, n)$

```
library(emdbook)
m <- mle2( y ~ dbetabinom( shape1=a , shape2=b , size=n  ) ,
  start=list(a=2,b=2) )
```