In [1]:
```
using CSV
using DataFrames
using Optim
using GLM
using Statistics
```

Maximum Likelihood Estimation

The maximum likelihood estimator is important because it uses all of the information in a fully specified statistical model.

- Asymptotically efficient
- Possible misspecification of the model

Suppose we have a sample of size n of the random vectors y and z. Suppose the joint density of $Y = \left(y_1, \ldots, y_n\right)$ and $Z = \left(z_1, \ldots, z_n\right)$ is characterized by a parameter vector psi_0:

$$f_{YZ}(Y, Z, \psi_0)$$

The **likelihood function** is just this density evaluated at other values of ψ:

$$L(Y, Z, \psi) = f(Y, Z, \psi), \quad \psi \in \Psi$$

where Ψ is a parameter space. The **maximum likelihood estimator**, ψ_0, is the value of psi that maximizes the likelihood function:

$$\psi_0 = \arg\max_{\psi \in \Psi} L(Y, Z, \psi)$$

Example (Count data)

Suppose we have a sample $Y = y_1, \ldots, y_n$ where the data are counts: the number of times some event occurs in a given interval of time, e.g., number of visits to the doctor a year. The simplest count data density is the Poisson:

$$f_Y(y; \lambda) = \frac{e^{-\lambda}\lambda^y}{y!}$$

If the observations are i.i.d. according to the above density, then the joint density of the sample is:

$$L = \prod_{i=1}^{n} \frac{e^{-\lambda}\lambda^{y_i}}{y_i!}$$

A little calculus and algebra shows that:

$$\lambda_0 = \bar{y} = \arg\max_{\lambda \in \Lambda} L$$

Exogenous variables

The likelihood function can be factored as

$$f_{YZ}(Y,Z,\psi) = f_{Y|Z}(Y|Z,\theta)f_Z(Z,\rho)$$

where θ are whatever elements of ψ that happen to enter in the conditional density, and ρ are the elements that enter into the marginal density.

If θ and ρ share no elements, then the maximizer of the conditional likelihood function $f_{Y|Z}(Y|Z,\theta)$ with respect to θ is the same as the maximizer of the overall likelihood function $f_{YZ}(Y,Z,\psi) = f_{Y|Z}(Y|Z,\theta)f_Z(Z,\rho)$, for the elements of ψ that corresponds to θ.

Z are said to be exogenous for estimation of θ, and may more conveniently work with the conditional likelihood function $f_{Y|Z}(Y|Z,\theta)$ for the purpose of estimating θ.

Factorization of the Likelihood Function

- If the n observations are independent, the likelihood function can be written as

$$L(Y|Z,\theta) = \prod_{i=1}^{n} f(y_i|z_i,\theta)$$

- If not, we can factor the likelihood into:

$$L(Y|Z,\theta) = f(y_n|y_{1,}y_2,\dots y_{n-1},Z,\theta)f(y_{n-1}|y_{1,}y_2,\dots y_{n-2},Z,\theta)\cdots f(y_2|y_1,Z,\theta)f(y_1|Z,\theta)$$

using

$$\underbrace{f(y_{1,}y_2,\dots y_{n-1},y_n|Z,\theta)}_{\text{joint}} = \underbrace{f(y_n|y_{1,}y_2,\dots y_{n-1},Z,\theta)}_{\text{conditional}}\underbrace{f(y_{1,}y_2,\dots y_{n-1}|Z,\theta)}_{\text{marginal}}$$

Example (Bernoulli trial)

Suppose that we are flipping a coin that may be biased, so that the probability of a heads may not be 0.5. Maybe we are interested in estimating the probability of heads. Let Y be a binary variable that equals 1 in case of heads and 0 in case of tails. The outcome of a coin toss ia a Bernoulli random variable:

$$f_Y(y,p) = p^y(1-p)^{1-y}$$

The average log-likelihood function is:

$$Q_n(p) = \frac{1}{n}\sum_{t=1}^{n} y_t \ln p + (1-y_t)\ln(1-p)$$

The derivative of a representative term is:

$$\begin{aligned}\frac{\partial \ln f_Y(y,p)}{\partial p} &= \frac{y}{p} - \frac{(1-y)}{(1-p)} \\ &= \frac{y-p}{p\,(1-p)}\end{aligned}$$

Hence,

$$\frac{\partial Q_n(p)}{\partial p} = \frac{1}{n}\sum_{t=1}^{n}\frac{y_t - p}{p\,(1-p)}.$$

Setting to zero and solving:

$$\hat{p} = \bar{y}$$

It's easy to calculate p using MLE in this case.

Let's verify that it is consistent:

- For a given p, the objective function converges to the limit of its expectation

$$Q_n(p) = \frac{1}{n}\sum_{t=1}^{n} y_t \ln p + (1-y_t)\ln(1-p) \to p_0 \ln p + (1-p_0)\ln(1-p)$$

- Compact parameter space: p_0 lies between 0 and 1
- Objective function continuous and measurable: as long as $p \in (0,1)$ the objective function is bounded

It is easy to see that p that maximizes

$$\frac{\partial Q_n}{\partial p} = \frac{p_0}{p} + \frac{(1-p_0)}{\ln(1-p)}$$

is p_0

Example (classical linear regression model)

Let's suppose that a dependent variable is normally distributed: $y \sim N(\mu_0, \sigma_0^2)$, so

$$f_y(y;\mu_0,\sigma_0^2) = \frac{1}{\sqrt{2\pi\sigma_0^2}}\exp\left(-\frac{(y-\mu_0)^2}{2\sigma_0^2}\right)$$

Suppose that the mean, μ_0, depends on some regressors, x. The simplest way to do this is to assume that $\mu_0 = x'\beta_0$. With this, the density, conditional on x is

$$f_y(y|x;\beta_0,\sigma_0^2) = \frac{1}{\sqrt{2\pi\sigma_0^2}}\exp\left(-\frac{(y-x'\beta_0)^2}{2\sigma_0^2}\right)$$

This is an example of **parameterization** of a density, making some parameters depend on additional variables and new parameters. With an i.i.d. sample of size n, the overall conditional density is the product of the conditional density of each observation:

$$f_y(y_1,y_2,\ldots,y_n|x_1,x_2,\ldots,x;\beta_0,\sigma_0^2) = \prod_{t=1}^{n}\frac{1}{\sqrt{2\pi\sigma_0^2}}\exp\left(-\frac{(y_t-x_t'\beta_0)^2}{2\sigma_0^2}\right)$$

Taking logarithms, and evaluating at some point in the parameter space, we get the log-likelihood function:

$$\ln L(Y|X;\beta,\sigma^2) = -n\ln\sqrt{2\pi} - n\ln\sigma - \sum_{t=1}^{n}\frac{\left(y_t - x_t'\beta\right)^2}{2\sigma^2}$$

- Observe that the first order conditions for β are the same for the OLS estimator.
- We know that the OLS estimator is consistent without making distributional assumptions regarding the errors. Thus, estimator for MLE will also be consistent.

Let's verify that:

In [3]:
```
n = 30

# random true parameters
theta = [1.0,-1.0,0.0,1.0]
b = theta[1:3]
sig = theta[4]

# generate random data
x = [ones(n,1) rand(n,2)]
e = sig*randn(n,1)
y = x*b + e

function normal(theta, y, x)
    b = theta[1:end-1]
    s = theta[end][1]
    e = (y - x*b)./s
    logdensity = -log.(sqrt.(2.0*pi)) .- 0.5*log(s.^2) .- 0.5*e.*e
end
function fminunc(obj, x; tol = 1e-08)
    results = Optim.optimize(obj, x, LBFGS(),
                             Optim.Options(
                             g_tol = tol,
                             x_tol=tol,
                             f_tol=tol))
    return results.minimizer, results.minimum, Optim.converged(results)
    #xopt, objvalue, flag = fmincon(obj, x, tol=tol)
    #return xopt, objvalue, flag
end

# do ML: note minus sign, also, do "edit(normal)" to see what's done
obj = theta -> -mean(normal(theta, y, x))
thetahat, junk, junk = fminunc(obj, theta)

# results
println("the true parameters: ", theta)
println("the ML estimates: ", thetahat)
println("the OLS estimates: ", inv(x'*x)*x'*y)
```

```
the true parameters: [1.0, -1.0, 0.0, 1.0]
the ML estimates: [1.152744593202836, -1.5264717345860088, 0.045387260254273504, 0.55411
040905102]
the OLS estimates: [1.152744593214728; -1.5264717308046245; 0.04538725994390768]
```

Consistency: general result

$$\mathcal{E}\left(\ln\left(\frac{L(\theta)}{L(\theta_0)}\right)\right) \leq \ln\left(\mathcal{E}\left(\frac{L(\theta)}{L(\theta_0)}\right)\right)$$

by Jensen's inequality ($\ln(\cdot)$ is a concave function).

Now, the expectation on the RHS is

$$\mathcal{E}\left(\frac{L(\theta)}{L(\theta_0)}\right) = \int \frac{L(\theta)}{L(\theta_0)} L(\theta_0) dy = 1,$$

since $L(\theta_0)$ \emph{is} the density function of the observations, and since the integral of any density is 1. \newpage Therefore, since $\ln(1) = 0$,

$$\mathcal{E}\left(\ln\left(\frac{L(\theta)}{L(\theta_0)}\right)\right) \leq 0,$$

or

$$\mathcal{E}\left(s_n\left(\theta\right)\right) - \mathcal{E}\left(s_n\left(\theta_0\right)\right) \leq 0$$

or

$$\mathcal{E}\left(s_n\left(\theta\right)\right) \leq \mathcal{E}\left(s_n\left(\theta_0\right)\right)$$

Taking limits of each side:

$$s_\infty(\theta) \leq s_\infty(\theta_0)$$

except on a set of zero probability.

So the true parameter value is the maximizer of the limiting objective function (we are in Case 1 of the three cases discussed above - a fully correctly specified model).

If the identification assumption holds, then there is a unique maximizer, so the inequality is strict if $\theta \neq \theta_0$:

$$s_\infty(\theta) < s_\infty(\theta_0), \forall \theta \neq \theta_0, \text{a.s.}$$

Therefore, θ_0 is the unique maximizer of $s_\infty(\theta)$, and thus,

$$\lim_{n\to\infty} \hat{\theta} = \theta_0, \text{ a. s.}$$

So, the ML estimator is consistent for the true parameter value. In practice, we will need to check identification for the specific model under consideration.

We can verify that by increasing the n in the above example.

In [3]:
```
n = 10000
# random true parameters
theta = [1.0,-1.0,0.0,1.0]
b = theta[1:3]
sig = theta[4]

# generate random data
```

```
x = [ones(n,1) rand(n,2)]
e = sig*randn(n,1)
y = x*b + e

function normal(theta, y, x)
    b = theta[1:end-1]
    s = theta[end][1]
    e = (y - x*b)./s
    logdensity = -log.(sqrt.(2.0*pi)) .- 0.5*log(s.^2) .- 0.5*e.*e
end
function fminunc(obj, x; tol = 1e-08)
    results = Optim.optimize(obj, x, LBFGS(),
                             Optim.Options(
                             g_tol = tol,
                             x_tol=tol,
                             f_tol=tol))
    return results.minimizer, results.minimum, Optim.converged(results)
    #xopt, objvalue, flag = fmincon(obj, x, tol=tol)
    #return xopt, objvalue, flag
end

# do ML: note minus sign, also, do "edit(normal)" to see what's done
obj = theta -> -mean(normal(theta, y, x))
thetahat, junk, junk = fminunc(obj, theta)

# results
println("the true parameters: ", theta)
println("the ML estimates: ", thetahat)
println("the OLS estimates: ", inv(x'*x)*x'*y)
```

```
the true parameters: [1.0, -1.0, 0.0, 1.0]
the ML estimates: [0.9805976694253452, -1.0025731879860325, 0.04297209517333844, 0.99809
01062153833]
the OLS estimates: [0.9805976694602873; -1.0025731882166724; 0.0429720953733152]
```

Asymptotic Normality: general result

You will go over with with TA on Friday :)