

Examples for the `lstbayes` package

Jeffrey B. Arnold

June 26, 2016

Some example programs typset using the `listings` language drivers provided by the `lstbayes` package.

1 BUGS

The Rats model from the OpenBUGS Examples Volume I: <http://www.openbugs.net/Examples/Rats.html>.

```
model {
  for( i in 1 : N ) {
    for( j in 1 : T ) {
      Y[i , j] ~ dnorm(mu[i , j], tau.c)
      mu[i , j] ← alpha[i] + beta[i] * (x[j] - xbar)
      culmative.Y[i , j] ← culmative(Y[i , j], Y[i , j])
      post.pv.Y[i , j] ← post.p.value(Y[i , j])
      prior.pv.Y[i , j] ← prior.p.value(Y[i , j])
      replicate.post.Y[i , j] ← replicate.post(Y[i , j])
      pv.post.Y[i , j] ← step(Y[i , j] - replicate.post.Y[i , j])
      replicate.prior.Y[i , j] ← replicate.prior(Y[i , j])
      pv.prior.Y[i , j] ← step(Y[i , j] - replicate.prior.Y[i , j])
    }
    alpha[i] ~ dnorm(alpha.c , alpha.tau)
    beta[i] ~ dnorm(beta.c , beta.tau)
  }
  tau.c ~ dgamma(0.001 , 0.001)
  sigma ← 1 / sqrt(tau.c)
  alpha.c ~ dnorm(0.0 , 1.0E-6)
  alpha.tau ~ dgamma(0.001 , 0.001)
  beta.c ~ dnorm(0.0 , 1.0E-6)
  beta.tau ~ dgamma(0.001 , 0.001)
  alpha0 ← alpha.c - xbar * beta.c
}
```

2 JAGS

Linear regression example from John Myles White, <http://www.johnmyleswhite.com/notebook/2010/08/20/using-jags-in-r-with-the-rjags-package/>.

```
model {
  for (i in 1:N){
    y[i] ~ dnorm(y.hat[i], tau)
    y.hat[i] ← a + b * x[i]
  }
  a ~ dnorm(0, .0001)
  b ~ dnorm(0, .0001)
  tau ← pow(sigma, -2)
  sigma ~ dunif(0, 100)
}
```

3 Stan

Rats example from https://github.com/stan-dev/example-models/blob/master/bugs_examples/vol1/rats/rats_vec.stan.

```
# http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/Vol1.pdf
```

```
# Page 3: Rats
```

```
data {
  int<lower=0> N;
  int<lower=0> T;
  real x[T];
  real y[N,T];
  real xbar;
}
transformed data {
  real x_minus_xbar[T];
  real y_linear[N*T];

  for (t in 1:T)
    x_minus_xbar[t] ← x[t] - xbar;

  for (n in 1:N)
    for (t in 1:T)
      y_linear[(n-1)*T + t] ← y[n, t];
}
parameters {
  real alpha[N];
  real beta[N];

  real mu_alpha;
```

```

real mu_beta;

real<lower=0> sigmasq_y;
real<lower=0> sigmasq_alpha;
real<lower=0> sigmasq_beta;
}
transformed parameters {
  real<lower=0> sigma_y;
  real<lower=0> sigma_alpha;
  real<lower=0> sigma_beta;

  sigma_y <- sqrt(sigmasq_y);
  sigma_alpha <- sqrt(sigmasq_alpha);
  sigma_beta <- sqrt(sigmasq_beta);
}
model {
  real pred[N*T];

  for (n in 1:N)
    for (t in 1:T)
      pred[(n-1)*T + t] <- fma(beta[n], x_minus_xbar[t], alpha[n]);

  mu_alpha ~ normal(0, 100);
  mu_beta ~ normal(0, 100);
  sigmasq_y ~ inv_gamma(0.001, 0.001);
  sigmasq_alpha ~ inv_gamma(0.001, 0.001);
  sigmasq_beta ~ inv_gamma(0.001, 0.001);
  alpha ~ normal(mu_alpha, sigma_alpha); // vectorized
  beta ~ normal(mu_beta, sigma_beta); // vectorized

  y_linear ~ normal(pred, sigma_y); // vectorized
}

```