Stan Models

Jason Doll

May 18, 2018

Exercise 1: Estimating Mean and SD

```
data{
  int<lower=0> n; //number of obserations
  vector[n] y;    //observations as a vector
}

parameters {
  real<lower=0> sigma;  //standard deviation to be estimated
  real mu;              //Mean to be estimated
}

model {

  //reference priors
  sigma ~ cauchy(0,5);
  mu ~ normal(0,100);

  //likelihood, loop through number of observations
  for(i in 1:n){
    y[i] ~ normal(mu, sigma);
  }

}
```

Exercise 2: Linear Regression

```
data {
  int<lower=0> n;  //number of observations
  vector[n] x;     //observed x values, predictors
  vector[n] y;     //observed y values, response
}

parameters {
  real<lower=0> sigma; //standard deviation
  real alpha;          //y-intercept
  real beta;           //slope
}

model {
  //reference priors
  alpha ~ normal(0,100);
  beta ~ normal(0,100);
  sigma ~ cauchy(0,5);

  //likelihood, loop through number of observations
  for(i in 1:n){
    y[i] ~ normal(alpha + beta * x[i], sigma);
  }
}
```

Exercise 3(a): Stan code for centered parameterization

```
parameters{
  real y;
  vector[9] x;
}
model {
  y~normal(0,3);
  x~normal(0,exp(y/2));
}
```

Exercise 3(b): Stan code for non-centered parameterization

```
parameters {
  real y_raw;
  vector[9] x_raw;
}
transformed parameters {
  real y;
  vector[9] x;
  y = 3.0 * y_raw;
  x = exp(y/2) * x_raw;
}
model {
  y_raw ~ normal(0, 1); // implies y ~ normal(0, 3)
  x_raw ~ normal(0, 1); // implies x ~ normal(0, exp(y/2))
}
```

Exercise 4(a): Stan code for centered parameterization

```
data{
  int<lower=0> n; //number of observations
  vector[n] y;    //observations as a vector
  vector[n] x;     //observed x values, predictors
  int gp[n];      //subject indicator
  int ngps;       //number of groups
}

parameters{
  real<lower=0> sigma_y;   //individual standard deviation to be estimated
  real<lower=0> sigma_mu;  //group standard deviation to be estimated
  real gpmu[ngps];         //Group mean
  real mu;                 //Global mean across all groups
  real beta;               //slope
}

model {
  //reference priors
  sigma_y ~ cauchy(0,5);
  sigma_mu ~ cauchy(0,5);
  beta ~ normal(0,100);
  mu ~ normal(0,100);

  for(k in 1:ngps){
    gpmu[k] ~ normal(mu,sigma_mu); //individual group means
  }

  //likelihood, loop through number of observations
  for(i in 1:n){
    y[i] ~ normal(gpmu[gp[i]] + beta * x[i], sigma_y);
  }
}
```

Exercise 4(b): Stan code for non-centered parameterization

```
data{
  int<lower=0> n; //number of obserations
  vector[n] y;    //observations as a vector
  vector[n] x;     //observed x values, predictors
  int gp[n];      //subject indicator
  int ngps;       //number of groups
}

parameters{
  real<lower=0> sigma_y;   //individual standard deviation to be estimated
  real<lower=0> sigma_mu;  //group standard deviation to be estimated
  real gpmu_raw[ngps];     //Group mean_raw
  real mu;                 //Global mean across all groups
  real beta;               //slope
}

transformed parameters{
  real gpmu[ngps];              //Group mean

  for(k in 1:ngps){
    gpmu[k] = mu + gpmu_raw[k] * sigma_mu;
  }
}

model {
  //reference priors
  sigma_y ~ cauchy(0,5);
  sigma_mu ~ cauchy(0,5);
  beta ~ normal(0,100);
  mu ~ normal(0,100);

  for(k in 1:ngps){
    gpmu_raw[k] ~ normal(0,1);//implies gpmu[k] ~ normal(mu,sigma_mu); //individual group mean
  }

  //likelihood, loop through number of observations
  for(i in 1:n){
    y[i] ~ normal(gpmu[gp[i]] + beta * x[i], sigma_y);
  }
}
```

Exercise 5(a): WAIC, Stan code for centered parameterization

```
data {
  int<lower=0> n;  //number of observations
  vector[n] x;     //observed x values, predictors
  vector[n] y;     //observed y values, response
}

parameters {
  real<lower=0> sigma; //standard deviation
  real alpha;          //y-intercept
  real beta;           //slope
}

model {

  //reference priors
  alpha ~ normal(0,100);
  beta ~ normal(0,100);
  sigma ~ cauchy(0,5);

  //likelihood, loop through number of observations
  for(i in 1:n){
    y[i] ~ normal(alpha + beta * x[i], sigma);
  }

}

generated quantities{
    vector[n] log_lik;

    for ( i in 1:n ) {
        log_lik[i] = normal_lpdf( y[i] | alpha + beta * x[i], sigma);
    }
}
```

Exercise 5(b): WAIC, Stan code for non-centered parameterization

```
data{
  int<lower=0> n; //number of obserations
  vector[n] y;    //observations as a vector
  vector[n] x;     //observed x values, predictors
  int gp[n];      //subject indicator
  int ngps;       //number of groups
}

parameters{
  real<lower=0> sigma_y;   //individual standard deviation to be estimated
  real<lower=0> sigma_mu;  //group standard deviation to be estimated
  real gpmu_raw[ngps];     //Group mean_raw
  real mu;                 //Global mean across all groups
  real beta;               //slope
}

transformed parameters{
  real gpmu[ngps];              //Group mean

  for(k in 1:ngps){
    gpmu[k] = mu + gpmu_raw[k] * sigma_mu;
  }
}

model {
  //reference priors
  sigma_y ~ cauchy(0,5);
  sigma_mu ~ cauchy(0,5);
  beta ~ normal(0,100);
  mu ~ normal(0,100);

  for(k in 1:ngps){
    gpmu_raw[k] ~ normal(0,1);//implies gpmu[k] ~ normal(mu,sigma_mu); //individual group mean
  }

  //likelihood, loop through number of observations
  for(i in 1:n){
    y[i] ~ normal(gpmu[gp[i]] + beta * x[i], sigma_y);
  }
}

generated quantities{
    vector[n] log_lik;

    for ( i in 1:n ) {
        log_lik[i] = normal_lpdf( y[i] | gpmu[gp[i]] + beta * x[i], sigma_y);
    }
}
```

Exercise 6: Iris data, Linear Regression

```
data {
  int<lower=0> n;  //number of observations
  vector[n] x;     //observed x values, predictors
  vector[n] y;     //observed y values, response
}

parameters {
  real<lower=0> sigma; //standard deviation
  real alpha;          //y-intercept
  real beta;           //slope
}

model {

  //reference priors
  alpha ~ normal(0,100);
  beta ~ normal(0,100);
  sigma ~ cauchy(0,5);

  //likelihood, loop through number of observations
  for(i in 1:n){
    y[i] ~ normal(alpha + beta * x[i], sigma);
  }

}
```