Regression Models

Stan supports regression models from simple linear regressions to multilevel generalized linear models.

Linear Regression

The simplest linear regression model is the following, with a single predictor and a slope and intercept coefficient, and normally distributed noise. This model can be written using standard regression notation as

$$y_n = \alpha + \beta x_n + \epsilon_n \quad \text{where} \quad \epsilon_n \sim \text{normal}(0, \sigma).$$

This is equivalent to the following sampling involving the residual,

$$y_n - (\alpha + \beta X_n) \sim \text{normal}(0, \sigma),$$

and reducing still further, to

$$y_n \sim \text{normal}(\alpha + \beta X_n, \sigma).$$

This latter form of the model is coded in Stan as follows. Links to source:

Option 1 R, Python, Stan

Option 2 regression_1.R, regression_1.py, regression_1.stan

Option 3 Folder

```
data {
  int<lower=0> N;
  vector[N] x;
  vector[N] y;
}
parameters {
  real alpha;
  real beta;
  real<lower=0> sigma;
}
model {
  y ~ normal(alpha + beta*x, sigma);
}
```

new way above with insert from dist, old way below with source typed into doc.

```
data {
  int<lower=0> N;
  vector[N] x;
  vector[N] y;
}
parameters {
  real alpha;
  real beta;
  real<lower=0> sigma;
}
model {
  y ~ normal(alpha + beta * x, sigma);
}
```

There are `N` observations, each with predictor `x[n]` and outcome `y[n]`. The intercept and slope parameters are `alpha` and `beta`. The model assumes a normally distributed noise term with scale `sigma`. This model has improper priors for the two regression coefficients.

Matrix Notation and Vectorization

The sampling statement in the previous model is vectorized, with

```
  y ~ normal(alpha + beta*x, sigma);
```

above, new way excerpts from source code on disk. Rmarkdown code that generates the snippet is:

```
lines = strsplit(stan_file,"\n")[[1]]
cat(paste(lines[12:12],collapse="\n"))
```

```
  y ~ normal(alpha + beta*x, sigma);
```

END