Bayesian one-way ANOVA with two levels using Stan

In this program, we assume that we have a vector of N observations that can be classified according to one of two levels of a factor.

Our objective is to fit a one-way ANOVA with two levels:

$$Y = \beta_1 X_1 + \beta_2 X_2 + e$$

where:

- Y is a vector of observed measurements (highway mpg values).
- β_1 is a parameter, assumed to be a random variable and given a prior distribution.
- X_1 is a binary predictor that is 1 if this y is categorized as level 1 of the factor, zero otherwise.
- β_2 is a parameter, assumed to be a random variable and given a prior distribution.
- X_2 is a binary predictor that is 1 if this y is categorized as level 2 of the factor, zero otherwise.
- e is a vector of independent normal random variables with mean zero and standard deviation σ_e.

In the Bayesian formulation, β_1, β_2 and σ_e are treated as parameters, which are random variables in the Bayesian framework.

The Bayes equation is:

$$f(\beta_1, \beta_2, \sigma_e | Y) \propto f(Y | \beta_1, \beta_2, \sigma_e) \cdot f(\beta_1) \cdot f(\beta_2) \cdot f(\sigma_e)$$

where:

- $f(\beta_1, \beta_2, \sigma_e | Y)$ is the (joint) posterior distribution of β_1, β_2 and σ_e given Y
- $f(Y | \beta_1, \beta_2, \sigma_e)$ is the conditional likelihood of Y given β_1, β_2 and σ_e
- $f(\beta_1)$ is the prior distribution for β_1
- $f(\beta_2)$ is the prior distribution for β_2
- $f(\sigma_e)$ is the prior distribution for σ_e

The likelihood and priors are specified in `one_way_ANOVA.stan`:

```
//single-factor ANOVA
data {
  int N;                        //sample size - all levels
  int L;                        //number of levels
  real y[N];                    //y values
  int lvl[N];                   //level for this observation
}
parameters {
  real alpha[L];                //Parameters for each level of the single factor
  real<lower=0> sigma_e;        //common error standard deviation for all levels
}
model {

  alpha ~ normal(0,100);        //normal priors for each alpha level
  sigma_e ~ cauchy(0,10);       //half-cauchy prior for error standard deviation

  for (i in 1:N){                           //loop through y values
    y[i] ~ normal(alpha[lvl[i]],sigma_e);  //y[j] ~ normal(alpha[grp],sigma_e)
  }
}
```

With this model file, we have the following specifications:

- At level 1 of the factor, the likelihood is $f(Y|\beta_1, \beta_2, \sigma_e)$ is $N(\beta_1, \sigma_e)$
- At level 2 of the factor, the likelihood is $f(Y|\beta_1, \beta_2, \sigma_e)$ is $N(\beta_2, \sigma_e)$
- The priors $f(\beta_i)$ are all $N(0, 100)$, coded as `beta` $\sim$ `normal(0,100)`
- The prior $f(\sigma_e)$ is half-cauchy (because of $< lower = 0 >$) coded as `sigma_e` $\sim$ `cauchy(0,10)`