データの解析方法の説明

計算した破損頻度(故障率)のばらつきを評価してその値がどの程度信頼できるかを表示する必要があります。

破損頻度(故障率)λは、以下のように簡単な計算で求まります。しかし、母集団の分布がどうなっているか、分散(ばらつき)がどの程度か判断できません。

$$\lambda = \frac{故障数}{集計運転時間}$$

そこで、本プラットフォームでは次の 2 つの方法で母集団の分布を推定、信頼区間を設定し信頼限界値を求め、破損頻度データのばらつき程度を表示します。

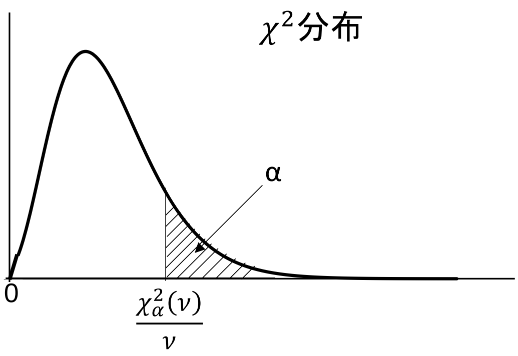
なお、下記の計算は基本的に同じものですが、2)では、破損頻度(故障率)のヒスとグラムを得ることができます。

1) χ(カイ)^2 分布による信頼区間の設定および信頼限界の計算値

2) ベイズ統計モデリング法(ソフト R-Stan)による信頼区間の設定および信頼限界の計算値

---

1) χ(カイ)^2 分布による信頼区間の設定および信頼限界の計算値



$$\frac{\chi^2_\alpha(\nu)}{\nu} : \int_{\chi^2_\alpha}^{\infty} \frac{1}{2\Gamma(\frac{\nu}{2})}\left(\frac{\chi^2}{2}\right)^{\frac{\nu}{2}-1} e^{-\chi^2/2} d\chi^2 = \alpha$$

図 1　χ^2 分布のパーセント点と自由度の比

χ^2分布のパーセント点と自由度の比を図 1 に示します。ここで、運転集計時間 τ の間で、n 回の破損があった場合、90%信頼区間は、以下で与えられます。

$$\left(\frac{1}{2\tau}z0.95,2n,\frac{1}{2\tau}z0.05,2(n+1)\right)$$

$z0.95,\nu$と $z0.05,\nu$は、自由度νを有するχ^2分布の上 95%, 5%を示します。

【計算例】

破損 n=6 回, 集計運転時間 τ =10,000 時間の場合、

故障率は、　　$\hat{\lambda}=\frac{n}{\tau}=6\cdot10^{-4}\,回/hr$

90%信頼区間は以下で示します。

$$\left(\frac{1}{2\tau}z0.95,2n,\frac{1}{2\tau}z0.05,2(n+1)\right)=\left(\frac{1}{20000}z0.95,12,\frac{1}{20000}z0.05,14\right)=(2.6\cdot10^{-4},11.8\cdot10^{-4})$$

従って、答は平均値=6.0E-04,　5%信頼限界値=2.6E-04, 95%信頼限界値=11.8E-04 になります。

【計算方法】　Excel 表でχ^2分布を表し信頼限界値を求める関数として以下を用います。

Excel 上でのχ^2分布関数

5%信頼限界値(下限値)=CHISQ.INV(0.05,2*n)/(2* τ)

95%信頼限界値(上限値)=CHISQ.INV(0.95,2*(n+1))/(2* τ)

なお、GFF プラットフォームにおけるデータ入力フォーマットでは、破損回数および集計運転時間を入力すると、破損頻度(故障率)および信頼限界値が自動的に計算・表示されるよう設定してあります。

2) ベイズ統計モデリング法(ソフト R-Stan)による信頼区間の設定および信頼限界の計算値

GFF を求める際に、母集団分布のパラメータがいくらになるのかよくわからない場合、定量化するために母集団分布のパラメータが従う確率分布を想定します。

Stan では　MCMC(マルコフ連鎖モンテカルロ法、ハミルトニアンモンテカルロ法)法を用いて任意の確率分布、事後分布に従う乱数を生成することができます。事後分布に従う乱数のヒストグラムを描けば、それが事後分布の形状となります。事後分布の平均値はMCMC 法により生成された「事後分布に従う乱数の平均値」を計算して求めます。

Stan による母集団分布の推定、平均と分散などの推定は以下のように行います。

1) 母集団分布の構造を決める(破損頻度の場合はポアソン分布に従う場合が多く、ポアソン分布を仮定し、信頼区間の設定から 5%あるいは 95%信頼限界を計算します)。
2) MCMC 法を使って、母集団のパラメータの事後分布に従う乱数を生成します。
3) 生成された乱数を使って、ベイズ更新を行って推定結果を解釈します。

```
data{
  int N;
  real T;
}
parameters{
  real<lower=0>lambda;
}
model{
  N~poisson(T*lambda);
}
```

```
library(rstan)
data<-list(N=2, T=1500300)
fit1<-stan'TEST1.stan',data=data)
lambda<-rstan::extract(fit1)$lambda
quantile(lambda,c(0.05,0.5,0.95))
stan_trace(fit1,pars='lambda')
stan_hist(fit1,pars='lambda')
stan_dens(fit1,pars='lambda',separate_chains=TRUE)
stan_ac(fit1,pars='lambda',separate_chains=TRUE)
```

図 2　R-Stan プログラムの一例と Stan を実行させるための R スクリプト

```
> lambda<-rstan::extract(fit1)$lambda
> quantile(lambda,c(0.05,0.5,0.95))
           5%          50%          95%
5.416048e-07 1.759193e-06 4.206821e-06
> stan_trace(fit1,pars='lambda')
> stan_hist(fit1,pars='lambda')
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
> stan_dens(fit1,pars='lambda',separate_chains=TRUE)
> stan_ac(fit1,pars='lambda',separate_chains=TRUE)
>
```

図 3　計算結果の出力の一部

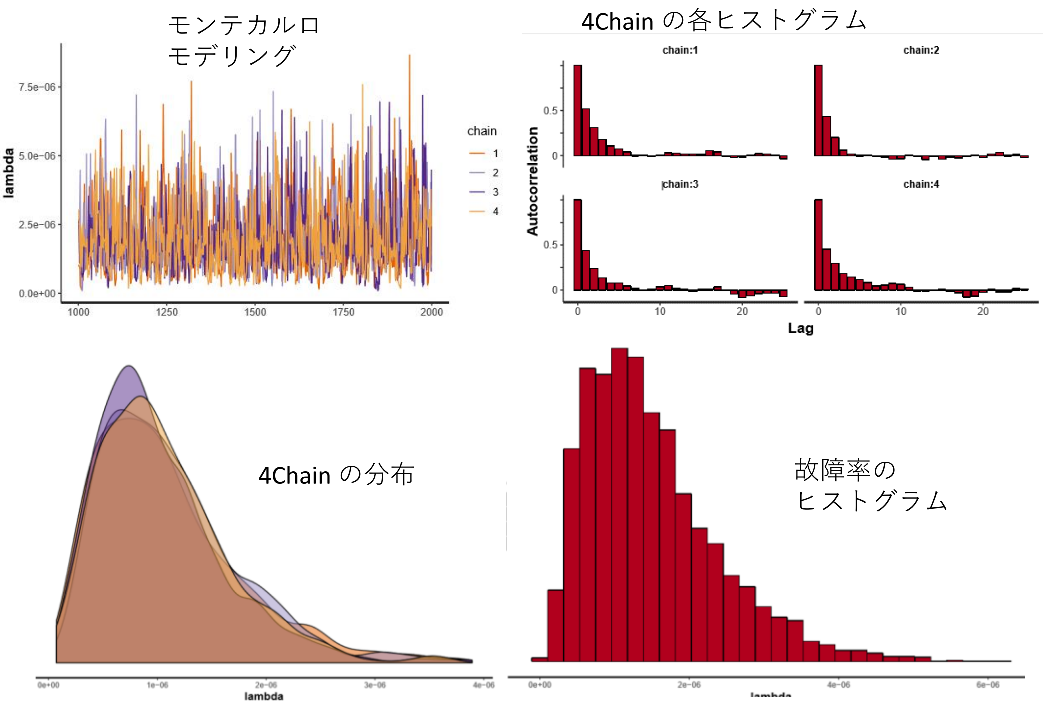


図 4　R-Stan によるベイス統計モデリングの解析計算結果図

図 2 に、R-Stan プログラムの一例と Stan を実行させるための R スクリプト、図 3 に計算結果の一例を示します。図 4 は、計算結果を図示したものです。2,000 回のモンテカルロモデリング、ベイズ更新(繰返し、4 Chains)を行って最終的に事後分布として故障率のヒストグラムを表示します。

以上のように、破損頻度(故障率)を平均値(50%)、下限値(5%信頼限界)、上限値(95%信頼限界)として表し、下限値と上限値の幅によりその精度を判断します。

以上