Bayesian Data Analysis - Assignment 6

November 6, 2017

1 Linear model: drowning data with Stan

x = year
y = number of people drown
$y \sim \mathcal{N}(\alpha + \beta x, \sigma^2)$

i)

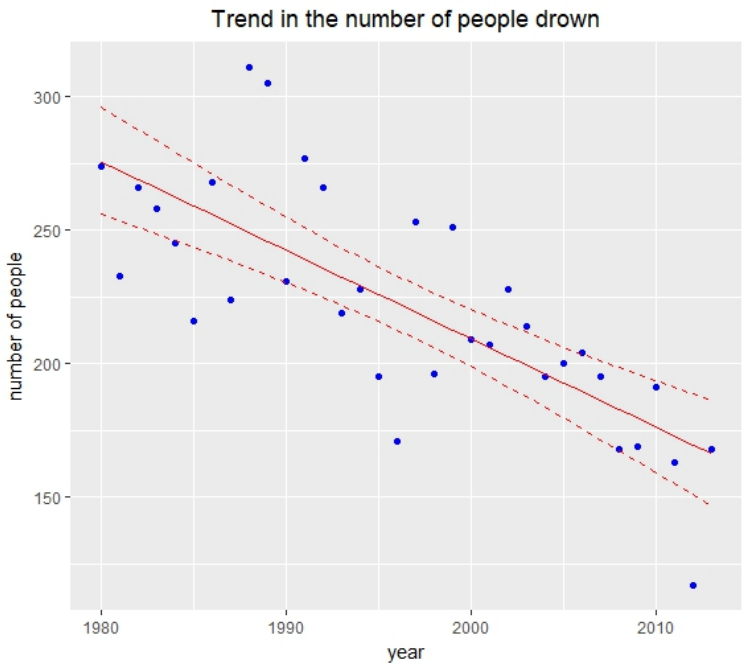

Figure 1: Trend in the number of people drown

From the Figure 1, we can conclude that the number of people drown per year decreases.

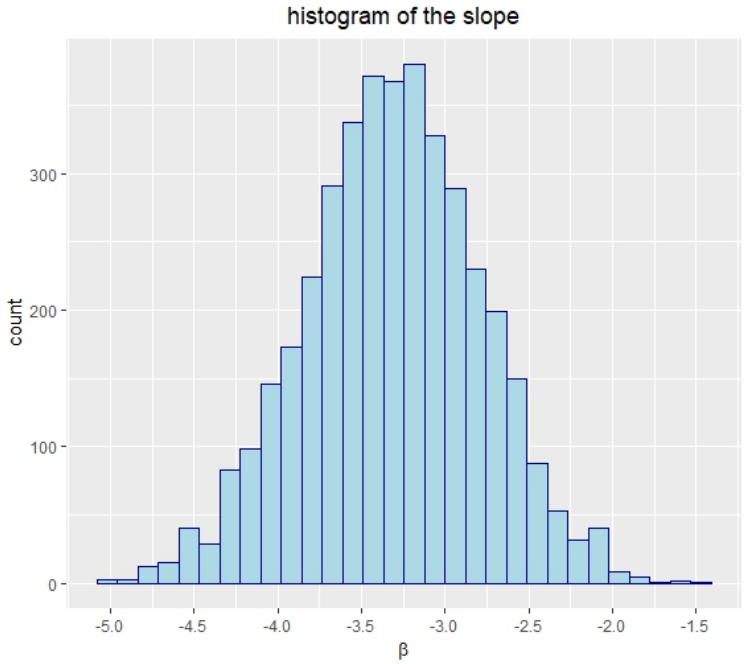

Figure 2: histogram of the slope

From the Figure 2, we can see that the slope are between [-6,-1]<0, showing that the trend is decreasing. The mean of the slope is around -3.31, the central-95% interval is [-4.35,-2.31].

ii)

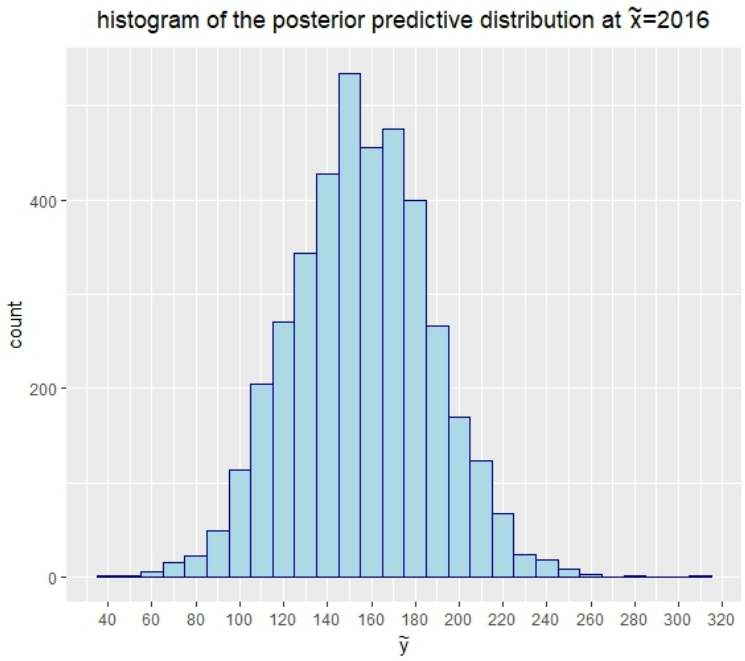

Figure 3: histogram of the posterior predictive distribution

$$\mu_{\tilde{y}} \approx 156$$

Stan:

```
data{
  int<lower=0> N; // number of data points
  vector[N] x; // time
  vector[N] y; // number of drownings
  real xpred; // input location for prediction
}
parameters{
  real alpha;
  real beta;
  real<lower=0> sigma;
}
transformed parameters{
  vector[N] mu;
```

```
  mu=alpha+beta*x;
}
model{
  y~normal(mu,sigma);
}
generated quantities{
  real ypred;
  vector[N] log_lik;
  ypred = normal_rng(alpha+beta*xpred,sigma);
  for(n in 1:N)
    log_lik[n]=normal_lpdf(y[n] | alpha + beta*x[n],sigma);
}
```

R code:

```
library("rstan")
library("ggplot2")
rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())
# import and organize data
raw_data<-read.table("drowning.txt")
drowning_data<-list(N=nrow(raw_data),
           x=raw_data$V1,
           y=raw_data$V2,
           xpred=2016
)

# fit Stan model
drowning_fit<-stan(file="drowning.stan",data=drowning_data)
drowning_result<-extract(drowning_fit,permuted=TRUE)
beta<-drowning_result$beta
mu<-drowning_result$mu
mu<-as.array(mu)

# show the trend
df_xy<-data.frame(x=drowning_data$x,y=drowning_data$y)

m_50<-c()
m_2p5<-c()
m_97p5<-c()
for(i in 1:length(drowning_data$x)){
```

```
  per_50 = quantile(mu[,i],0.5)
  per_2p5=quantile(mu[,i],0.025)
  per_97p5=quantile(mu[,i],0.975)
  m_50<-c(m_50,per_50)
  m_2p5<-c(m_2p5,per_2p5)
  m_97p5<-c(m_97p5,per_97p5)
}
m_50<-as.array(m_50)
m_2p5<-as.array(m_2p5)
m_97p5<-as.array(m_97p5)
df_mu<-data.frame(x=drowning_data$x,y1=m_50,y2=m_2p5,y3=m_97p5)

ggplot(df_xy,aes(df_xy$x))+
  geom_point(aes(y=df_xy$y),color="blue")+
  geom_line(aes(y=df_mu$y1),color="red")+
  geom_line(aes(y=df_mu$y2),linetype="dashed",color="red")+
  geom_line(aes(y=df_mu$y3),linetype="dashed",color="red")+
  labs(title="Trend in the number of people drown",
       x="year",y="number of people")+
  theme(plot.title = element_text(hjust = 0.5))

# plot histogram related to beta
beta_data<-data.frame(x=beta)
ggplot(data=beta_data,aes(x=beta_data))+
  geom_histogram(color="darkblue",,fill="lightblue")+
  scale_x_continuous(breaks=seq(-6,0,0.5))+
  labs(title="histogram of the slope",x=expression(beta))+
  theme(plot.title = element_text(hjust = 0.5))
mean(beta)
quantile(beta,c(0.025,0.975))

# plot histogram related to pred_y
ypred<-drowning_result$ypred
pred_y<-data.frame(x=ypred)
ggplot(data=pred_y,aes(x=pred_y))+
geom_histogram(bins=100,binwidth=10,color="darkblue",,fill="lightblue")+
scale_x_continuous(breaks=seq(0,320,20))+
labs(title=expression(paste("histogram of the posterior predictive distribution at ",
                            tilde(x), "=2016")),x=expression(tilde(y)))+
  theme(plot.title = element_text(hjust = 0.5))
```

```
mean(ypred)
```

2 Hierarchical model: factory data with Stan

(The R code is in the end of the report)

2.1 separate model

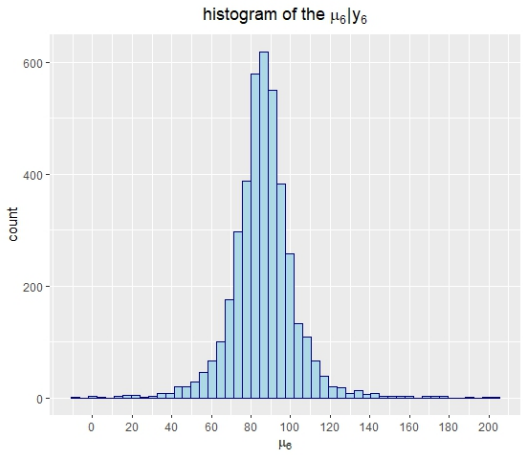

Figure 4: histogram of the $\mu_6 \mid y_6$

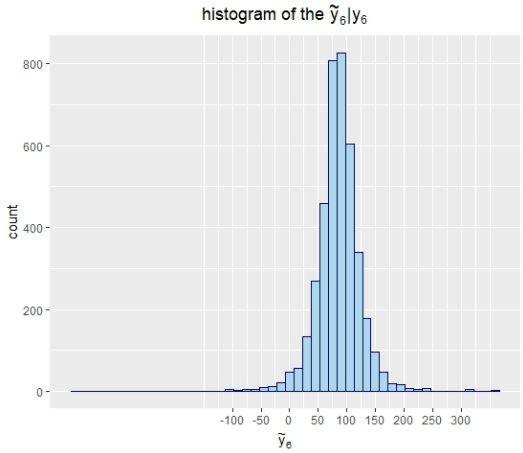

Figure 5: histogram of the $\tilde{y}_6 \mid y_6$

i) The mean of μ_6 is around 86.16, with the central 95%-interval: [58.69, 121.07].
ii) The mean of $\tilde{y}_6$ is around 85.13, with the central 95%-interval: [4.92, 160.63].
iii) Each machine has its own model, and we don't have any data related to the 7th machine, so we can not work out the posterior distribution of μ_7.

Stan:

```
data {
  int<lower=0> N; // number of data points
  int<lower=0> K; // number of groups
  int<lower=1,upper=K> x[N]; // group indicator
  vector[N] y; // measurements
}
parameters {
  vector[K] mu;              // group means
  vector<lower=0>[K] sigma; // group stds
}
model {
  y ~ normal(mu[x], sigma[x]);
}
generated quantities {
  real ypred;
  ypred = normal_rng(mu[6],sigma[6]);
}
```

2.2 pooled model

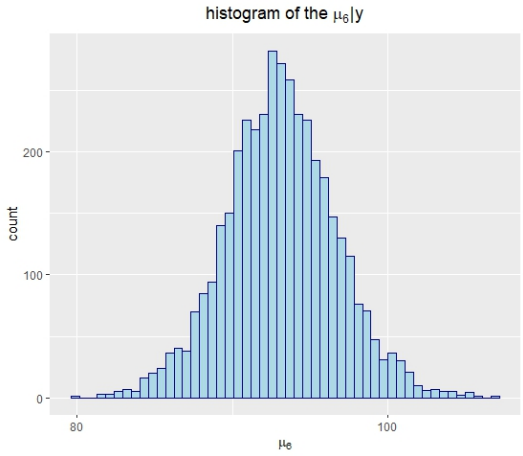

Figure 6: histogram of the $\mu_6 \mid y$

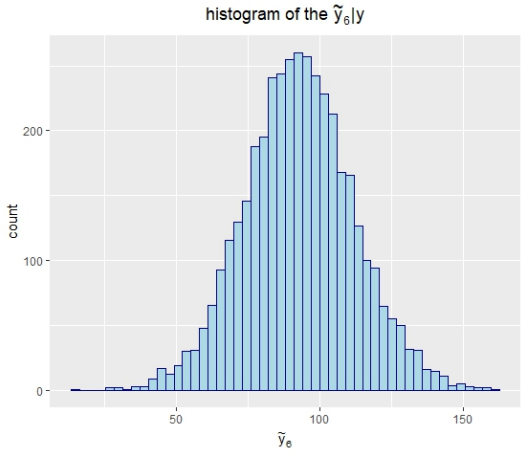

Figure 7: histogram of the $\tilde{y}_6 \mid y$

i) The mean of μ_6 is around 93.10, with the central 95%-interval: [85.95, 100.39].
ii) The mean of $\tilde{y}_6$ is around 92.94, with the central 95%-interval: [55.36, 132.21].
iii) all the machines share the same model, so the posterior distribution of

μ_7 is the same as μ_6 .

Stan:

```
data {
  int<lower=0> N; // number of data points
  int<lower=0> K; // number of groups
  int<lower=1,upper=K> x[N]; // group indicator
  vector[N] y; // measurements
}
parameters {
  real mu;    // common mean
  real sigma; // common std
}
model {
  y ~ normal(mu, sigma);
}
generated quantities {
  real ypred;
  ypred = normal_rng(mu,sigma);
}
```

2.3 hierarchical model

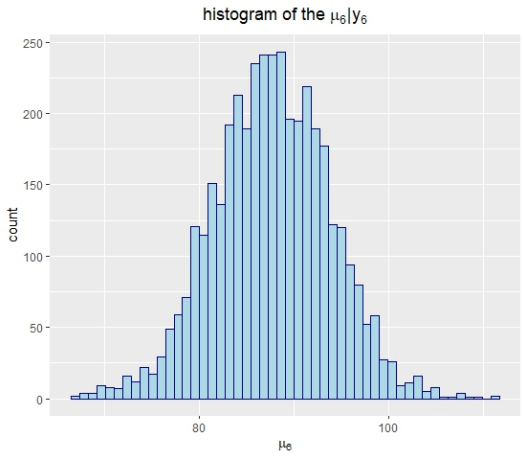

Figure 8: histogram of the $\mu_6 \mid y_6$

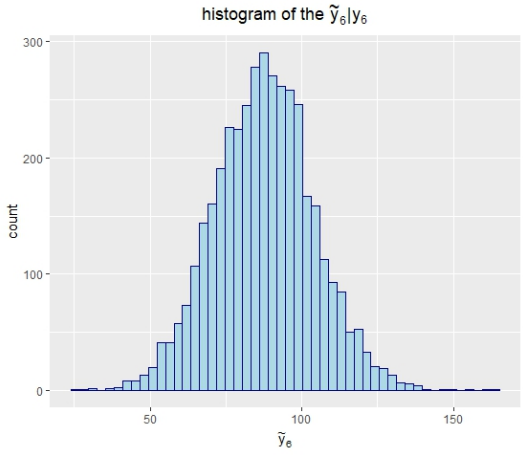

Figure 9: histogram of the $\tilde{y}_6 \mid y_6$

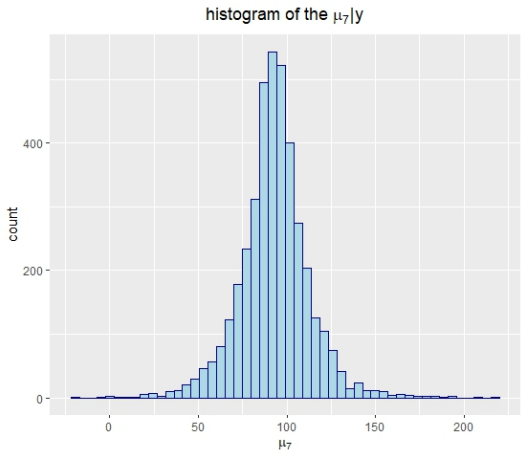

Figure 10: histogram of the $\mu_7 \mid y$

i) The mean of μ_6 is around 87.68, with the central 95%-interval: [75.50, 99.40].
ii) The mean of $\tilde{y}_6$ is around 87.62, with the central 95%-interval: [55.14 120.33].
iii) The mean of μ_7 is around 92.85, with the central 95%-interval: [51.45, 131.94].

Stan:

```
data {
  int<lower=0> N; // number of data points
  int<lower=0> K; // number of groups
  int<lower=1,upper=K> x[N]; // group indicator
  vector[N] y; // measurements
}
parameters {
  real mu0;              // prior mean
  real<lower=0> sigma0; // prior std
  vector[K] mu;              // group means
  real<lower=0> sigma; // common std
}
model {
  mu~normal(mu0,sigma0);
  y ~ normal(mu[x], sigma);
}
generated quantities {
  real ypred;
  real mu7_pred;
  ypred = normal_rng(mu[6],sigma);
  mu7_pred = normal_rng(mu0,sigma0);
}
```

R code:

```
library("rstan")
library("ggplot2")
rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())
# import and organize data
raw_data<-read.table("factory.txt")
factory_data<-list(N = ncol(raw_data)*nrow(raw_data),
    K = ncol(raw_data),
    x = rep(1:ncol(raw_data),nrow(raw_data)),
    y = c(t(raw_data[,1:ncol(raw_data)]))
)

# separate model
sf_fit<-stan(file="separate_factory.stan",data=factory_data)
sf_result<-extract(sf_fit,permuted=TRUE)
```

```
# mu
mu<-sf_result$mu
mu_df<-data.frame(x=mu)
ggplot(data=mu_df,aes(x=mu_df$x.6))+
  geom_histogram(color="darkblue",,fill="lightblue",bins=50)+
  scale_x_continuous(breaks=seq(0,200,20))+
  labs(title=expression(paste("histogram of the ",
                              mu[6],"|",y[6])),x=expression(mu[6]))+
  theme(plot.title = element_text(hjust = 0.5))
mean(mu_df$x.6)
quantile(mu,c(0.025,0.975))

# ypred
ypred<-sf_result$ypred
pred_y<-data.frame(x=ypred)
ggplot(data=pred_y,aes(x=pred_y))+
  geom_histogram(color="darkblue",,fill="lightblue",bins=50)+
  scale_x_continuous(breaks=seq(-100,300,50))+
  labs(title=expression(paste("histogram of the ",
                              tilde(y)[6],"|",y[6])),x=expression(tilde(y)[6]))+
  theme(plot.title = element_text(hjust = 0.5))
mean(ypred)
quantile(ypred,c(0.025,0.975))


# pooled model
pf_fit<-stan(file="pooled_factory.stan",data=factory_data)
pf_result<-extract(pf_fit,permuted=TRUE)

# mu
mu<-pf_result$mu
mu_df<-data.frame(x=mu)
ggplot(data=mu_df,aes(x=mu_df))+
  geom_histogram(color="darkblue",,fill="lightblue",bins=50)+
  scale_x_continuous(breaks=seq(0,200,20))+
  labs(title=expression(paste("histogram of the ",
                              mu[6],"|",y)),x=expression(mu[6]))+
  theme(plot.title = element_text(hjust = 0.5))
mean(mu_df$x)
quantile(mu,c(0.025,0.975))
```

```
# ypred
ypred<-pf_result$ypred
pred_y<-data.frame(x=ypred)
ggplot(data=pred_y,aes(x=pred_y))+
  geom_histogram(color="darkblue",,fill="lightblue",bins=50)+
  scale_x_continuous(breaks=seq(-100,300,50))+
  labs(title=expression(paste("histogram of the ",
                              tilde(y)[6],"|",y)),x=expression(tilde(y)[6]))+
  theme(plot.title = element_text(hjust = 0.5))
mean(ypred)
quantile(ypred,c(0.025,0.975))

# hierarchical model
hf_fit<-stan(file="hierarchical_factory.stan",data=factory_data)
hf_result<-extract(hf_fit,permuted=TRUE)

# mu
mu<-hf_result$mu
mu_df<-data.frame(x=mu)
ggplot(data=mu_df,aes(x=mu_df$x.6))+
  geom_histogram(color="darkblue",,fill="lightblue",bins=50)+
  scale_x_continuous(breaks=seq(0,200,20))+
  labs(title=expression(paste("histogram of the ",
                              mu[6],"|",y[6])),x=expression(mu[6]))+
  theme(plot.title = element_text(hjust = 0.5))
mean(mu_df$x.6)
quantile(mu_df$x.6,c(0.025,0.975))

# ypred
ypred<-hf_result$ypred
pred_y<-data.frame(x=ypred)
ggplot(data=pred_y,aes(x=pred_y))+
  geom_histogram(color="darkblue",,fill="lightblue",bins=50)+
  scale_x_continuous(breaks=seq(-100,300,50))+
  labs(title=expression(paste("histogram of the ",
                              tilde(y)[6],"|",y[6])),x=expression(tilde(y)[6]))+
  theme(plot.title = element_text(hjust = 0.5))
mean(ypred)
quantile(ypred,c(0.025,0.975))
```

```
# the 7th machine
mu7_pred<-hf_result$mu7_pred
pred_mu7<-data.frame(x=mu7_pred)
ggplot(data=pred_mu7,aes(x=pred_mu7))+
  geom_histogram(color="darkblue",,fill="lightblue",bins=50)+
  scale_x_continuous(breaks=seq(-100,300,50))+
  labs(title=expression(paste("histogram of the ",
                              mu[7],"|",y)),x=expression(mu[7]))+
  theme(plot.title = element_text(hjust = 0.5))
mean(mu7_pred)
quantile(mu7_pred,c(0.025,0.975))
```