

STAT 8700 Homework 4

Brian Detweiler

Friday, September 23, 2016

1. Consider data from a Normal population with unknown mean μ and variance σ^2 . A random sample of 100 observations is taken from this population, and the sample mean and variance were calculated to be 50 and 25 respectively.

(a) If we choose to use a $N - Inv - \chi^2(40, 0.64, 1, 16)$ prior distribution, write down the corresponding posterior distribution.

We are given the following:

$$\begin{aligned}n &= 100 \\ \bar{y} &= 50 \\ s^2 &= 25 \\ \mu_0 &= 40 \\ \kappa_0 &= 25 \\ \nu_0 &= 1 \\ \sigma_0^2 &= 16\end{aligned}$$

Likelihood:

$$p(y_i|\alpha, \beta, n_i, x_i) \propto [\text{logit}^{-1}(\alpha + \beta x_i)]^{y_i} [1 - \text{logit}^{-1}(\alpha + \beta x_i)]^{n_i - y_i}$$

Now we can use these values to calculate the joint posterior distribution, $N - Inv - \chi^2(\mu_n, \sigma_n^2/\kappa_n; \nu_n, \sigma_n^2)$:

$$\begin{aligned}\mu_n &= \frac{\kappa_0}{\kappa_0 + n} \mu_0 + \frac{n}{\kappa_0 + n} \bar{y} \\ &= \frac{25}{25 + 100} 40 + \frac{100}{25 + 100} 50 = 48 \\ \kappa_n &= \kappa_0 + n \\ &= 25 + 100 = 125 \\ \nu_n &= \nu_0 + n \\ &= 1 + 100 = 101 \\ \nu_n \sigma_n^2 &= \nu_0 \sigma_0^2 + (n - 1) s^2 + \frac{\kappa_0 n}{\kappa_0 + n} (\bar{y} - \mu_0)^2 \\ &= 1(16) + (100 - 1)25 + \frac{25(100)}{25 + 100} (50 - 40)^2 = 4491 \\ \sigma_n^2 &\approx 44.4653465347\end{aligned}$$

And thus our joint posterior distributin is $N - Inv - \chi^2(48, 0.355722772278; 101, 44.4653465347)$.

■

(b) Either analytically or via simulation, construct 95% credible intervals for σ^2 and μ .

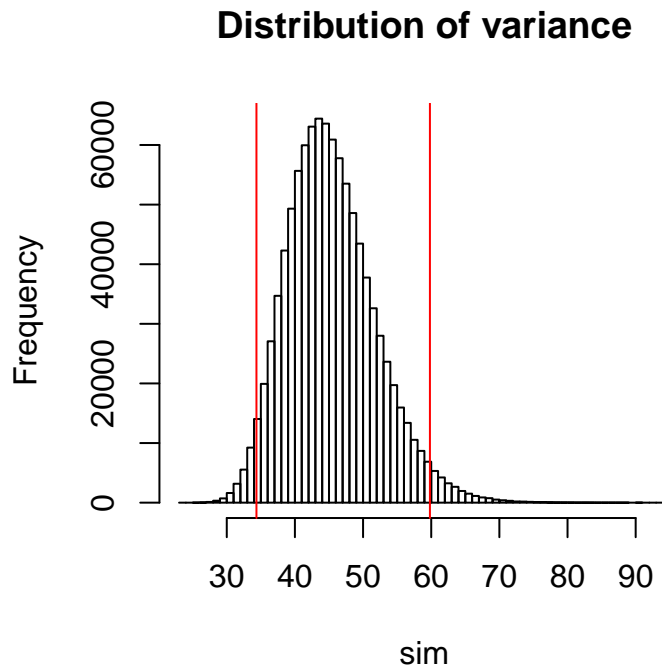
To simulate this, we first draw σ^2 from its marginal posterior distribution, $\sigma^2|y \sim \text{Inv} - \chi^2(\nu_n, \sigma_n^2)$

```
library(geoR)

set.seed(124)

x <- seq(0, 100, by = 0.001)
nu_n <- 101
sigma_n_2 <- 44.4653465347

sim <- rinvchisq(n = 1000000, df = nu_n, scale = sigma_n_2)
hist(sim, breaks = 90, main = 'Distribution of variance')
lower <- sort(sim)[25000]
upper <- sort(sim)[975000]
abline(v=lower, col='red')
abline(v=upper, col='red')
```



A 95% credible interval for σ^2 is (34.3564188, 59.8085225).

Then we sample from $N\left(\frac{\frac{\kappa_0}{\sigma^2}\mu_0 + \frac{n}{\sigma^2}\bar{y}}{\frac{\kappa_0}{\sigma^2} + \frac{n}{\sigma^2}}, \frac{1}{\frac{\kappa_0}{\sigma^2} + \frac{n}{\sigma^2}}\right)$ using the previous values for σ^2 .

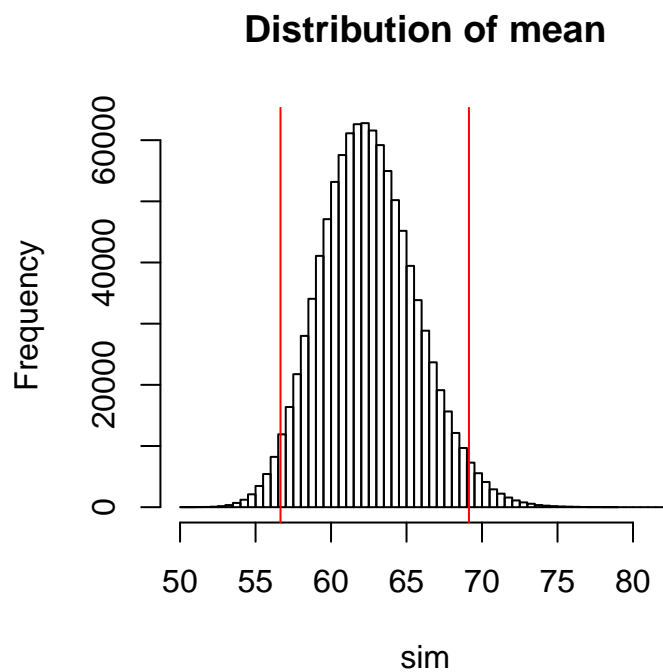
```
sigma_2 <- sim
kappa_0 <- 25
mu_0 <- 40
```

```

n <- 100
y_bar <- 50
mu_n <- ((kappa_0 / sigma_2) * mu_0) + ((n / sigma_2) * y_bar) / ((kappa_0 / sigma_2) + (n / sigma_2))
sigma_2_kappa_n <- 1 / ((kappa_0 / sigma_2) + (n / sigma_2))
sim <- rnorm(n = 1000000, mu_n, sigma_2_kappa_n)
hist(sim, breaks = 90, main = 'Distribution of mean')

lower <- sort(sim)[25000]
upper <- sort(sim)[975000]
abline(v=lower, col='red')
abline(v=upper, col='red')

```



A 95% credible interval for μ is (56.6518857, 69.1340067).

■

2. Two random variables are said to have a bivariate normal distribution with parameters, $\mu_U, \mu_V, \sigma_U^2, \sigma_V^2$, and ρ if they have the following density function:

$$f(u, v) = \frac{1}{2\pi\sigma_U\sigma_V\sqrt{1-\rho^2}} e^{-\frac{1}{2(1-\rho^2)} \left[\frac{(u-\mu_U)^2}{\sigma_U^2} + \frac{(v-\mu_V)^2}{\sigma_V^2} - \frac{2\rho(u-\mu_U)(v-\mu_V)}{\sigma_U\sigma_V} \right]}$$

where μ_U and σ_U^2 are the mean and variance of U , μ_V and σ_V^2 are the mean and variance of V , and ρ is the correlation between U and V .

Replace the uniform prior on α and β in the analysis of the bioassay by a bivariate normal prior with $\alpha \sim \text{Normal}(0, 4)$, $\beta \sim \text{Normal}(10, 100)$, and $\text{corr}(\alpha, \beta) = 0.5$. Repeat all the computations and plots discussed in section 3.7 and in class.

The book does a quick maximum likelihood of α and β .

```
x <- c(-0.86, -0.3, -0.05, 0.73)
n <- c(5, 5, 5, 5)
y <- c(0, 1, 3, 5)
y.mod <- c(0.5, 1, 3, 4.5)

data <- cbind(y, n - y)

fit <- glm(data ~ x, family = binomial)

coef(fit)
```

```
## (Intercept)          x
##  0.8465802    7.7488172
```

```
# This is the approximate covariance matrix:
summary(fit)$cov.unscaled
```

```
##              (Intercept)          x
## (Intercept)    1.038535    3.545987
## x              3.545987    23.743865
```

```
a <- coef(fit)[[1]]
b <- coef(fit)[[2]]
```

The maximum likelihood estimate for α is 0.8465802 and for β is 7.7488172.

```

bivariate_normal <- function(u, v) {
  mu_U <- 0
  mu_V <- 10

  sigma_2_U <- 4
  sigma_2_V <- 100

  sigma_U <- sqrt(sigma_2_U)
  sigma_V <- sqrt(sigma_2_V)

  rho <- 0.5

  first <- 1 / (2 * pi * sigma_U * sigma_V * sqrt(1 - rho^2))

  second <- exp(-(1 / (2 * (1 - rho^2)))
    * ( ((u - mu_U)^2 / sigma_2_U)
      + ((v - mu_V)^2 / sigma_2_V)
      - (2 * rho * (u - mu_U) * (v - mu_V)) / (sqrt(sigma_2_U) * sqrt(sigma_2_V))))
  rval <- first * second
  return(rval)
}

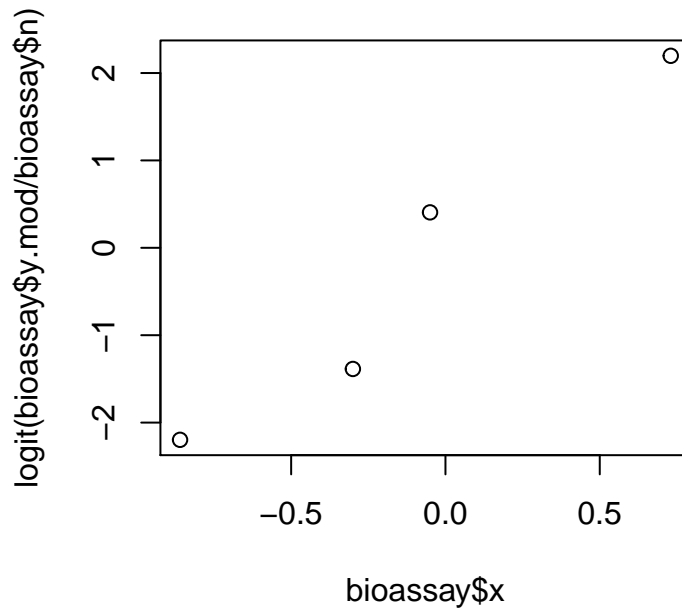
bioassay <- data.frame(cbind(x, n, y, y.mod))

logit <- function(x) {
  log(x / (1-x))
}

inv_logit <- function(x){
  exp(x) / (1 + exp(x))
}

plot(bioassay$x, logit(bioassay$y.mod / bioassay$n))

```



```
posterior <- function(a, b) {
  temp <- 1
  x <- bioassay$x
  y <- bioassay$y
  n <- bioassay$n
  for (i in 1:length(x)) {
    temp <- (temp * (inv_logit(a + (b * x[i]))^y[i])
              * ((1 - inv_logit(a + (b * x[i]))^(n[i] - y[i]))))
  }

  bivariate_normal(a, b) * temp
}

posterior_contour <- function(alpha_min,
                              alpha_max,
                              grid_size_alpha,
                              beta_min,
                              beta_max,
                              grid_size_beta,
                              drawlabels = TRUE) {

  # Generate a list of alpha values
  alpha <- seq(alpha_min, alpha_max, length = grid_size_alpha)

  # Generate a list of beta values
  beta <- seq(beta_min, beta_max, length = grid_size_beta)

  # Evaluate the posterior density and all possible combinations of alpha and beta values.
  post.dens <- outer(alpha, beta, 'posterior')
```

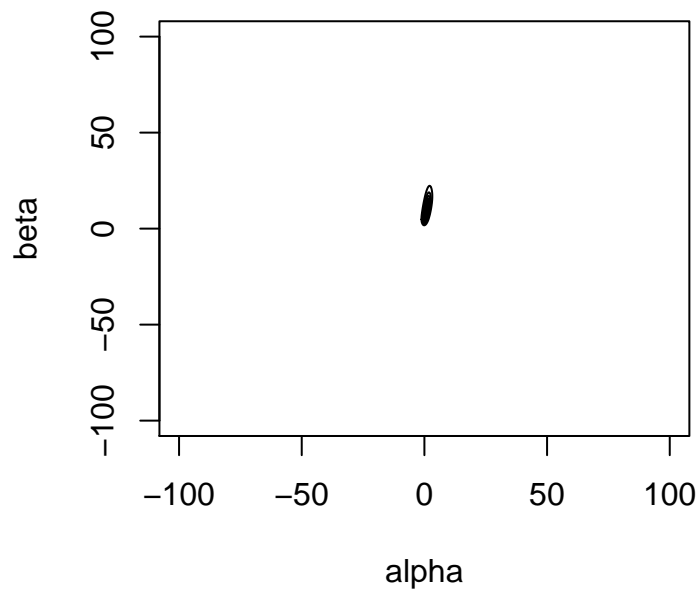
```

# Rescale the posterior so values are now relative to height of posterior mode.
scaled.dens <- post.dens / max(post.dens)

# Draw contour plot.
contour(alpha,
        beta,
        scaled.dens,
        levels = c(0.05, 0.15, 0.25, 0.35, 0.45, 0.55, 0.65, 0.75, 0.85, 0.95),
        xlab="alpha",
        ylab="beta",
        drawlabels = drawlabels)
}

posterior_contour(-100, 100, 2001, -100, 100, 2001)

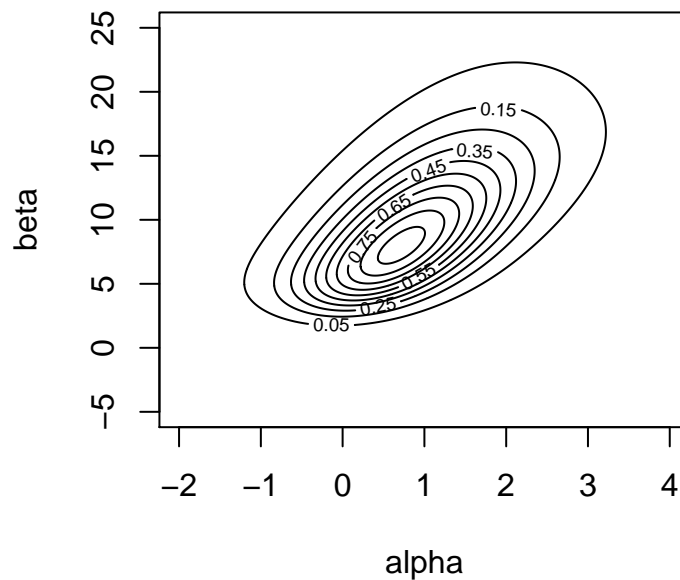
```



```

posterior_contour(-2, 4, 2001, -5, 25, 2001)

```



```
##Sampling from the joint posterior

posterior_sample <- function(alpha_min,
                              alpha_max,
                              grid_size_alpha,
                              beta_min,
                              beta_max,
                              grid_size_beta,
                              nsim) {

  # Generate a list of alpha values
  alpha <- seq(alpha_min, alpha_max, length = grid_size_alpha)

  # Generate a list of beta values
  beta <- seq(beta_min, beta_max, length = grid_size_beta)

  # Evaluate the posterior density and all possible combinations of alpha and beta values.
  post.dens <- outer(alpha, beta, 'posterior')

  # Rescale the posterior so values are now relative to height of posterior mode.
  scaled.dens <- post.dens / max(post.dens)

  # Makes sure our discrete probability density sums to 1
  normalized.dens = scaled.dens / sum(scaled.dens)

  marg.alpha <- rowSums(normalized.dens)

  #Sample 'nsim' values of alpha and beta
```



```

# Define a vector to hold our alpha values
sampled.alpha <- NULL

#Define a vector to hold out beta values
sampled.beta <- NULL

alpha_width <- (alpha_max - alpha_min) / (grid_size_alpha - 1)

beta_width <- (beta_max-beta_min) / (grid_size_beta - 1)

for (i in 1:nsim) {

  #Generate a value of alpha from the marginal of alpha
  temp.alpha <- sample(alpha, 1, marg.alpha, replace = TRUE)

  # Translate the obtained alpha to the appropriate row of our grid
  j <- ((temp.alpha - alpha_min) / alpha_width) + 1

  # Calculate the condition distribution of beta, conditional on the obtained value of alpha
  conditional.beta <- normalized.dens[j, ] / sum(normalized.dens[j, ])

  # Obtain a beta from the conditional distribution of beta
  temp.beta <- sample(beta, 1, conditional.beta, replace = TRUE)

  # Add some noise to try to undiscretize the posterior
  temp.alpha <- temp.alpha + runif(1, -alpha_width / 2, alpha_width / 2)

  temp.beta <- temp.beta + runif(1, -beta_width / 2, beta_width / 2)

  # Add the generated alpha to the list of previously generated alpha
  sampled.alpha <- c(sampled.alpha, temp.alpha)

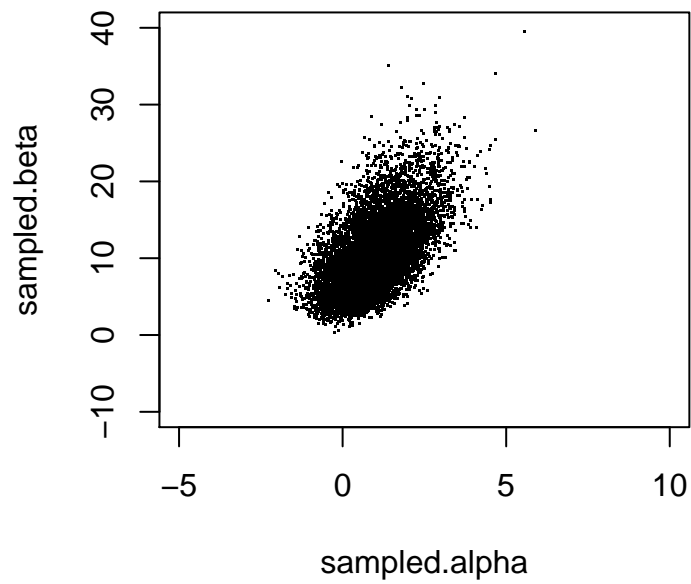
  sampled.beta<-c(sampled.beta, temp.beta)
}

data.frame(sampled.alpha, sampled.beta)
}

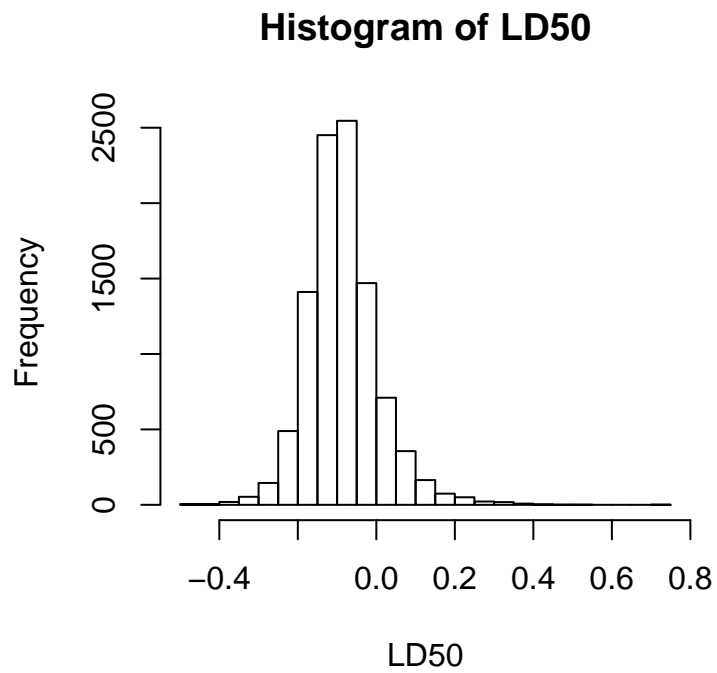
sims <- posterior_sample(-5, 10, 2001, -10, 40, 2001, 10000)

plot(sims, xlim = c(-5, 10), ylim = c(-10, 40), pch='.')

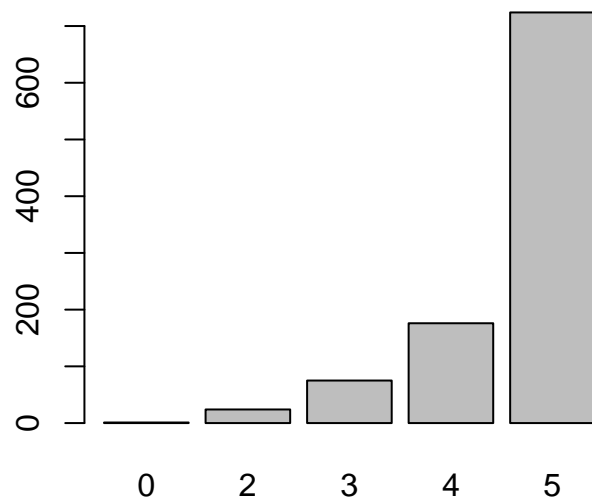
```



```
# Posterior Distribution of LD50 # LD50 is the dose at which the death rate is 50%, only makes sense for  
LD50 <- (-sims$sampled.alpha[sims$sampled.beta > 0] / sims$sampled.beta[sims$sampled.beta > 0])  
  
hist(LD50, br = 35)
```



```
# Posterior Predictive distribution for a dose level  $x = 0.25$ ,  $n = 5$   
#  $y \sim \text{Bin}(5, \theta)$   
#  $\text{logit}(\theta) = \alpha + \beta x$   
 $\theta \leftarrow \text{inv\_logit}(\text{sims}\$sampled.\alpha + (\text{sims}\$sampled.\beta * 0.25))$   
 $y \leftarrow \text{rbinom}(1000, 5, \theta)$   
 $\text{barplot}(\text{table}(y))$ 
```



■

3. Consider the airline fatalities data discussed in the previous exercise. Let us suppose that we now assume that the number of fatal accidents in year t follows a Poisson distribution with mean $\alpha + \beta t$.

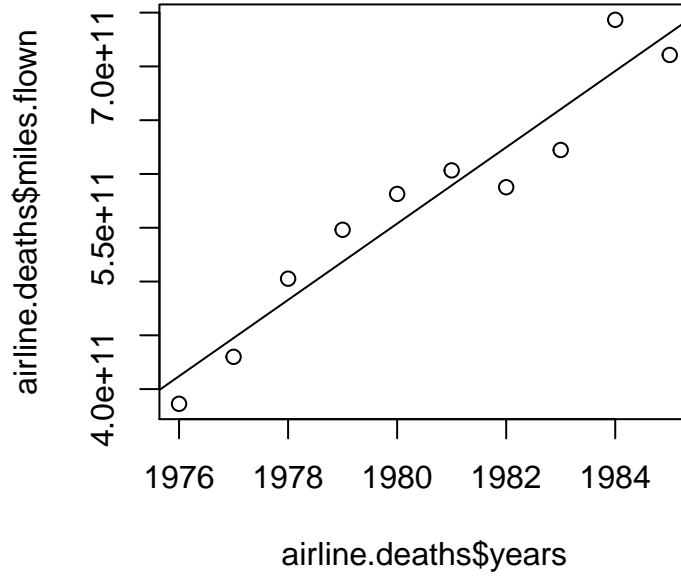
```
years <- c(1976:1985)
fatal.accidents <- c(24, 25, 31, 31, 22, 21, 26, 20, 16, 22)
passenger.deaths <- c(734, 516, 754, 877, 814, 362, 764, 809, 223, 1066)
death.rate <- c(0.19, 0.12, 0.15, 0.16, 0.14, 0.06, 0.13, 0.13, 0.03, 0.15)

airline.deaths <- as.data.frame(cbind(years, fatal.accidents, passenger.deaths, death.rate))
miles.flown <- (airline.deaths$passenger.deaths / airline.deaths$death.rate) * 100000000
airline.deaths$miles.flown <- miles.flown

airline.deaths
```

##	years	fatal.accidents	passenger.deaths	death.rate	miles.flown
## 1	1976	24	734	0.19	386315789474
## 2	1977	25	516	0.12	430000000000
## 3	1978	31	754	0.15	502666666667
## 4	1979	31	877	0.16	548125000000
## 5	1980	22	814	0.14	581428571429
## 6	1981	21	362	0.06	603333333333
## 7	1982	26	764	0.13	587692307692
## 8	1983	20	809	0.13	622307692308
## 9	1984	16	223	0.03	743333333333
## 10	1985	22	1066	0.15	710666666667

```
plot(airline.deaths$miles.flown ~ airline.deaths$years)
abline(lm(airline.deaths$miles.flown ~ airline.deaths$years))
```



(a) If we let y_t represent the number of fatal accidents in year t , write down $p(y_t|\alpha, \beta)$ the likelihood for year t in terms of the parameters α , and β .

$$t \sim \text{Poisson}(\alpha + \beta t)$$

$$p(y_t|\alpha, \beta) = \frac{(\alpha + \beta t)^{y_t}}{y_t!} e^{-(\alpha + \beta t)}$$

(b) If we assume uniform priors on α and β , write the posterior density for (α, β) .

With a uniform prior, we have $p(\alpha, \beta) \propto 1$. So the posterior is proportional to the likelihood.

We scale the years and let $t = 1$ represent 1975, let $t = 2$ represent 1976, and so on.

$$p(\alpha, \beta|y) \propto p(\alpha, \beta)p(y|\alpha, \beta)$$

$$\propto p(\alpha, \beta) \prod_{i=1}^{10} p(y_i|\alpha, \beta)$$

$$\propto \prod_{i=1}^{10} p(y_i|\alpha, \beta)$$

$$\propto \prod_{i=1}^{10} (\alpha + \beta t)^{y_t} e^{-(\alpha + \beta t)}$$

(c) Following the same idea as the boassay example (and the previous question) create a grid of possible α and β values on which to evaluate the joint posterior and plot the contours. Start with large ranges for α and β and refine based on the contour plot. Include all your iterations in your answer, not just your final grid and contour plot.

First, we create our posterior probability density,

```
posterior <- function(a, b) {
  rval <- 1
  y <- airline.deaths$fatal.accidents
  for (t in 1:length(y)) {
    # Note: The actual posterior is proportional to ((a + (b * t))^y[t] * exp(-(a + (b * t) ))))
    # But if we ignore NaNs produced by dpois (extremely large denominators by large factorials),
    # we get identical results
    rval <- (rval * dpois(y[t], lambda = (a + (b * t))))
  }

  return(rval)
}
```

```
posterior_contour <- function(alpha_min,
                              alpha_max,
                              grid_size_alpha,
                              beta_min,
                              beta_max,
                              grid_size_beta,
                              drawlabels = TRUE) {

  # Generate a list of alpha values
  alpha <- seq(alpha_min, alpha_max, length = grid_size_alpha)

  # Generate a list of beta values
  beta <- seq(beta_min, beta_max, length = grid_size_beta)

  # Evaluate the posterior density and all possible combinations of alpha and beta values.
  post.dens <- outer(alpha, beta, 'posterior')

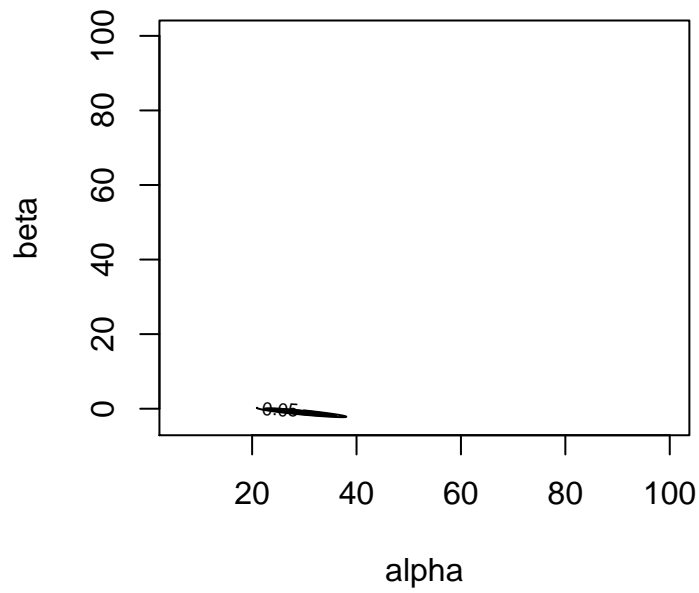
  # Set NaNs to 0
  post.dens[is.nan(post.dens)] <- 0

  # Rescale the posterior so values are now relative to height of posterior mode.
  scaled.dens <- post.dens / max(post.dens)

  # Draw contour plot.
  contour(alpha,
          beta,
          scaled.dens,
          levels = c(0.05, 0.15, 0.25, 0.35, 0.45, 0.55, 0.65, 0.75, 0.85, 0.95),
          xlab="alpha",
          ylab="beta",
          drawlabels = drawlabels)
}
```

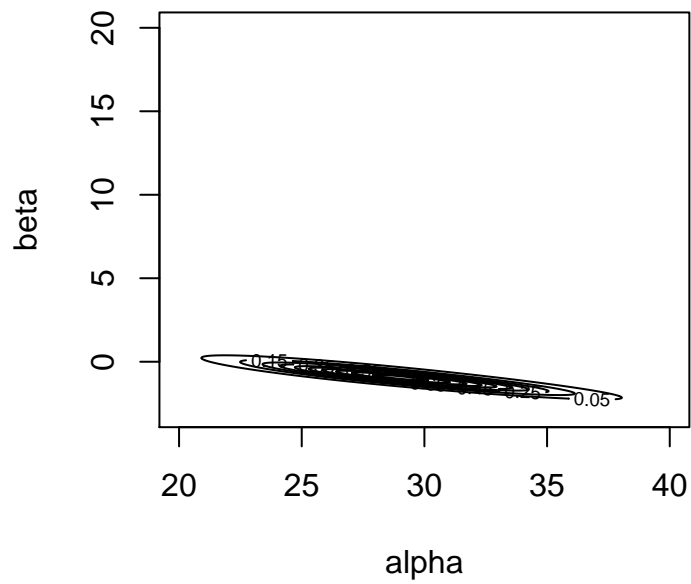
Then plot the contours

```
posterior_contour(6, 100, 2001, -3, 100, 2001)
```



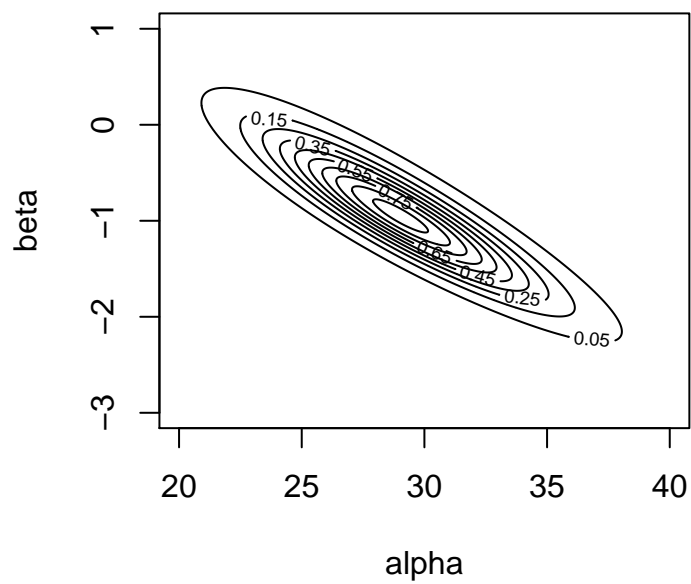
We can zoom in further,

```
posterior_contour(20, 40, 2001, -3, 20, 2001)
```

And now, we know just about where the ranges are, so we'll zoom in on those,

```
posterior_contour(20, 40, 2001, -3, 1, 2001)
```



(d) Simulate 100,000 values of α and β from the joint posterior and plot the histogram of the posterior density of the expected number of fatal accidents in 1986, $\alpha + 1986\beta$.

```
##Sampling from the joint posterior

posterior_sample <- function(alpha_min,
                             alpha_max,
                             grid_size_alpha,
                             beta_min,
                             beta_max,
                             grid_size_beta,
                             nsim) {

  # Generate a list of alpha values
  alpha <- seq(alpha_min, alpha_max, length = grid_size_alpha)

  # Generate a list of beta values
  beta <- seq(beta_min, beta_max, length = grid_size_beta)

  # Evaluate the posterior density and all possible combinations of alpha and beta values.
  post.dens <- outer(alpha, beta, 'posterior')
  # Set NANs to 0
  post.dens[is.nan(post.dens)] <- 0

  # Rescale the posterior so values are now relative to height of posterior mode.
  scaled.dens <- post.dens / max(post.dens)

  # Makes sure our discrete probability density sums to 1
  normalized.dens <- scaled.dens / sum(scaled.dens)

  marg.alpha <- rowSums(normalized.dens)

  #Sample 'nsim' values of alpha and beta

  # Define a vector to hold our alpha values
  sampled.alpha <- NULL

  #Define a vector to hold out beta values
  sampled.beta <- NULL

  alpha_width <- (alpha_max - alpha_min) / (grid_size_alpha - 1)

  beta_width <- (beta_max - beta_min) / (grid_size_beta - 1)

  for (i in 1:nsim) {

    #Generate a value of alpha from the marginal of alpha
    temp.alpha <- sample(alpha, 1, marg.alpha, replace = TRUE)

    # Translate the obtained alpha to the appropriate row of our grid
    j <- ((temp.alpha - alpha_min) / alpha_width) + 1
  }
}
```

```

    # Calculate the condition distribution of beta, conditional on the obtained value of alpha
    conditional.beta <- normalized.dens[j, ] / sum(normalized.dens[j, ])

    # Obtain a beta from the conditional distribution of beta
    temp.beta <- sample(beta, 1, conditional.beta, replace = TRUE)

    # Add some noise to try to undiscretize the posterior
    temp.alpha <- temp.alpha + runif(1, -alpha_width / 2, alpha_width / 2)

    temp.beta <- temp.beta + runif(1, -beta_width / 2, beta_width / 2)

    # Add the generated alpha to the list of previously generated alpha
    sampled.alpha <- c(sampled.alpha, temp.alpha)

    sampled.beta <- c(sampled.beta, temp.beta)
  }

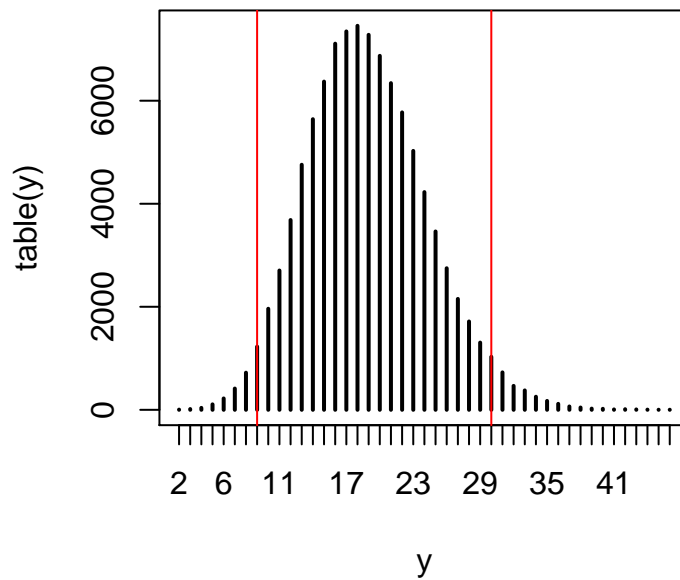
  data.frame(sampled.alpha, sampled.beta)
}

sims <- posterior_sample(15, 40, 2001, -3, 2, 2001, 100000)

# plot(sims, xlim = c(15, 40), ylim = c(-3, 2), pch='.')
y <- rpois(100000, lambda = (sims$sampled.alpha + sims$sampled.beta * 11))
plot(table(y))

q025 <- sort(y)[2500]
q975 <- sort(y)[97500]
abline(v = q025, col='red')
abline(v = q975, col='red')

```



(e) Use your simulated values of α and β to simulate the number of fatal accidents in 1986. Use your simulations to construct a 95% predictive (credible) interval.

(9, 30) is a 95% credible interval for the number of fatal accidents in 1986.

(f) Return to your simulated values of β , calculate (well, estimate) $P(\beta < 0)$, that is, the probability that the number of fatal accidents per year is decreasing.

```
prob.decreasing <- length(sims$sampled.beta[sims$sampled.beta < 0]) / length(sims$sampled.beta)
```

There is a 95.829% chance the number of fatal accidents is decreasing.

■