PSTAT 115 Homework 1

Aaron Barel / Kevin Ayala

October 2, 2018

1

(a) The population of interest is all of the words in all of JK Rowling's Harry Potter books. The population quantity of interest is how many times the word "dark" is used in each chapter. The sampling units are the randomly selected chapters.

(b) Frequency of word counts can be a good estimand for writing style. That is count how many times a certain word was used and divide it by the total number of words used. This may give a good estimand for the sentiment. To measure sentiment we can tokenize the words and assign it values that gauge the sentiment/feeling of the word. However, words that are oftenly used but give no sentiment to the context of the writing such as words like "the" should be filtered out to give an even better sentiment analysis. In general, depending on what you want to examine, the total word count should be for relative words for the metric you are trying to understand.

(c) $L(\lambda) = p(y|\lambda) = \prod_{i=1}^{n} \frac{\lambda^{y_i} e^{-n\lambda}}{y_i!}$By simplifying we get the likelihood $L(\lambda)$ as follows:$L(\lambda) = \frac{\lambda^{\sum_{i=1}^{n} y_i} e^{-n\lambda}}{\prod_{i=1}^{n} y_i!}$

(d) Taking the log-likelihood we get $l(\lambda) = \ln(L(\lambda)) = (\sum_{i=1}^{n} y_i)\ln(\lambda) - n\lambda - \sum_{i=1}^{n} \ln(y_i!)$. By taking the partial derivative in respect to λ and setting to 0 we get $\frac{\partial l}{\partial \lambda} = \frac{\sum_{i=1}^{n} y_i}{\lambda} - n \to \frac{\sum_{i=1}^{n} y_i}{\lambda} - n = 0$. Solving for λ we get the MLE: $\hat{\lambda} = \bar{Y}$

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------- tidyverse 1.2.1 --
```

```
## ✓ ggplot2 3.0.0     ✓ purrr   0.2.5
## ✓ tibble  1.4.2     ✓ dplyr   0.7.6
## ✓ tidyr   0.8.1     ✓ stringr 1.3.1
## ✓ readr   1.1.1     ✓ forcats 0.3.0
```

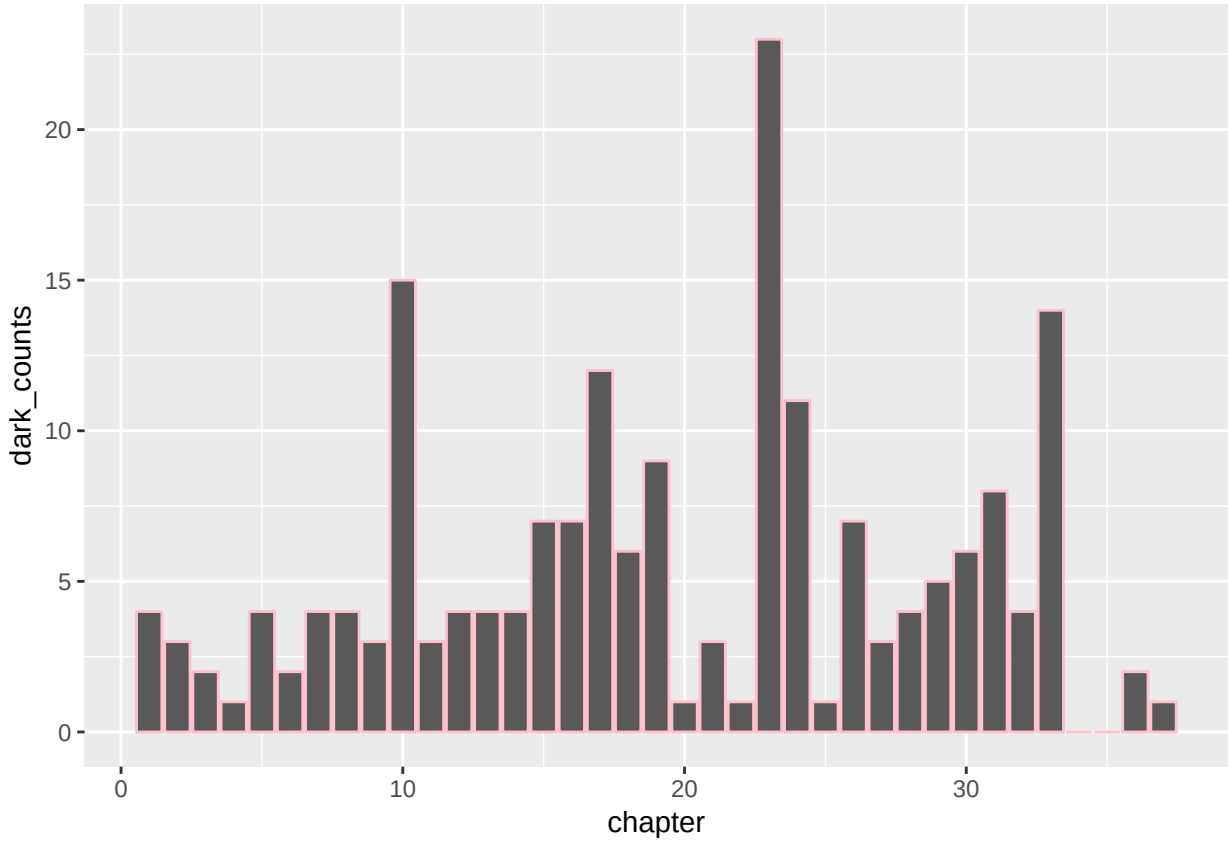
```
## -- Conflicts ------------------------------------------------------------------- tidyverse_conflicts() -
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(stringr)
library(tidytext)
library(harrypotter)

text_tb <- tibble(chapter = seq_along(deathly_hallows),
                  text = deathly_hallows)
tokens <- text_tb %>% unnest_tokens(word, text)
word_counts <- tokens %>% group_by(chapter) %>%
  count(word, sort = TRUE) %>% ungroup

word_counts_mat <- word_counts %>% spread(key =word, value = n, fill = 0)
```

```
dark_counts <- word_counts_mat$dark
```

(e)

```
df_darkCounts <- data.frame(dark_counts, chapter = text_tb$chapter)

ggplot(df_darkCounts, aes(x = chapter, y = dark_counts)) +
  geom_bar(stat = 'identity', color = 'pink')
```

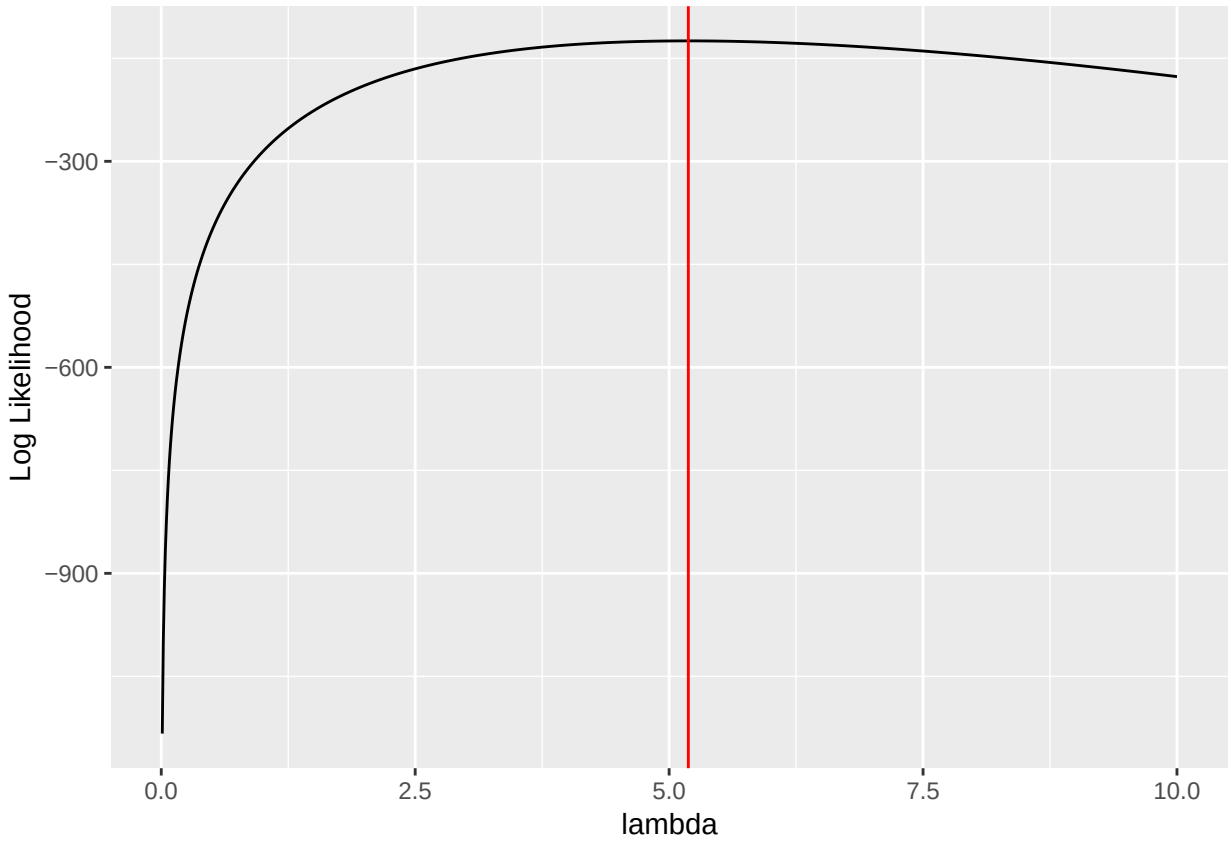


(f)

```
lambda = seq(0.01,10, by = 0.01)
n = length(df_darkCounts$chapter)

log_likelihood = sum(dark_counts)*log(lambda) - n*lambda - rep(sum(log(factorial(dark_counts))), each =

lambda_mle = mean(dark_counts)

lambda_mle
```

```
## [1] 5.189189
```

```
df_logLikelihood = data.frame(lambda = lambda, logLikelihood = log_likelihood)
ggplot(df_logLikelihood, aes(x = lambda, y = logLikelihood)) +
  geom_line() + geom_vline(xintercept = lambda_mle, color = "red") + labs(x = 'lambda', y = 'Log Likeli
```



The MLE for λ is 5.189189

2

```
chapter_lengths <- word_counts %>% group_by(chapter) %>%
  summarize(chapter_length = sum(n)) %>%
  ungroup %>% select(chapter_length) %>% unlist %>% as.numeric
```

(a) Since ν_i is the total length of the chapter, meaning the total amount of words, then $\frac{\nu_i}{1000}$ can be interpreted as per 1000 words. The interpretation of λ remains the same, it is the rate at which "dark" appears per chapter. So the interpretation of $\lambda * \frac{\nu_i}{1000}$ can be interpreted as the rate at which "dark" appears every 1000 words per chapter.

(b)

	Known	Unknown
Constants	$y_1, \ldots, y_n$, n, ν_i	λ
Variables	$Y_1, \ldots, Y_n$	

(c)

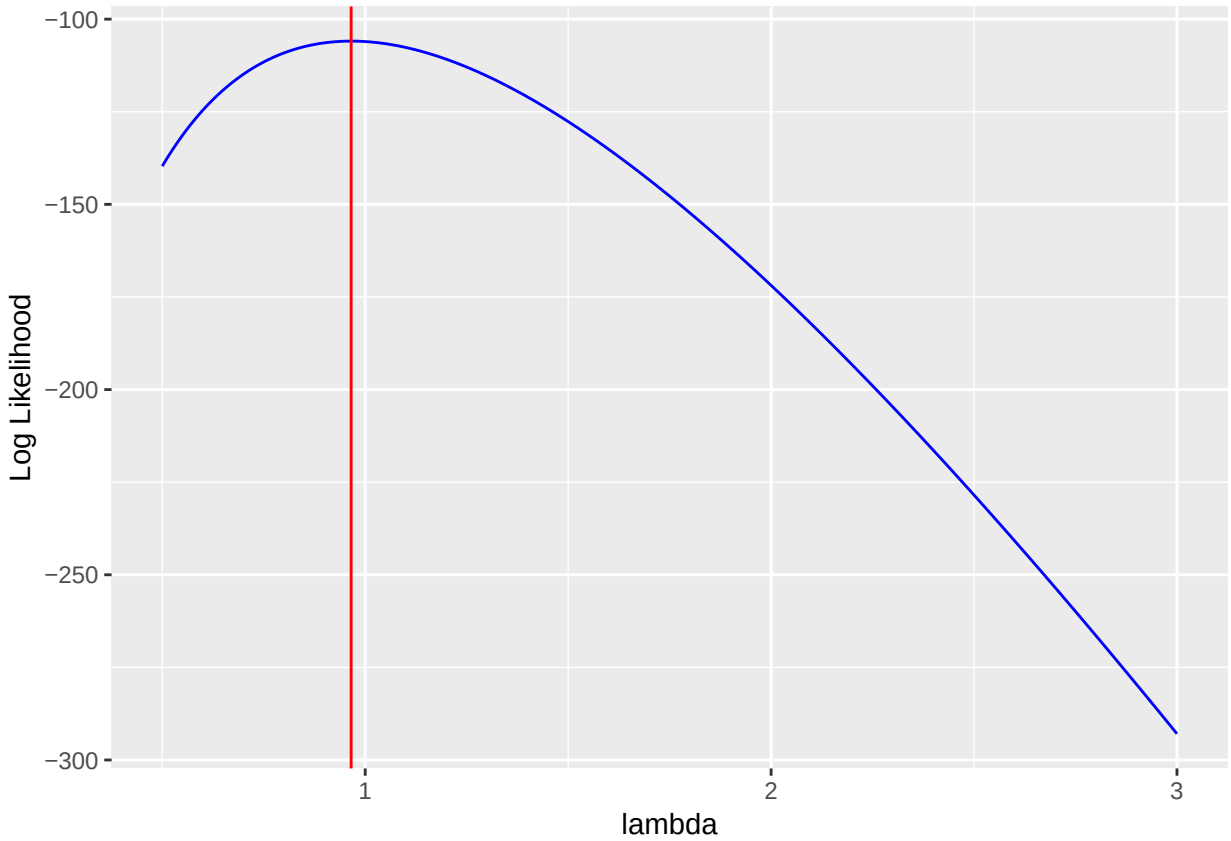
Let $\beta_i = \frac{\lambda \nu_i}{1000}$, then $L(\beta_i) = \prod_{i=1}^{n} \frac{e^{-\beta_i} \beta_i^{y_i}}{y_i!} = \frac{e^{\sum_{i=1}^{n} -\beta_i} \prod_{i=1}^{n} \beta_i^{y_i}}{\prod_{i=1}^{n} y_i!}$ By taking the log-likelihood we get $l(\beta_i) = -\sum_{i=1}^{n} \beta_i +$

$\sum_{i=1}^{n} y_i \ln(\beta_i) - \prod_{i=1}^{n} y_i!$. Here we can substitue back in $\frac{\lambda\nu_i}{1000}$ for β_i and we get $-\frac{\lambda}{1000}\sum_{i=1}^{n}\nu_i + \sum_{i=1}^{n} y_i \ln(\frac{\lambda\nu_i}{1000}) - \prod_{i=1}^{n} y_i!$.

Now we take the partial derivative in respect to λ and simplifying we get $\frac{\partial l}{\partial \lambda} = -\frac{\sum_{i=1}^{n}\nu_i}{1000} + \frac{\sum_{i=1}^{n} y_i}{\lambda}$. Setting $\frac{\partial l}{\partial \lambda} = 0$ and solving for λ we get $\hat{\lambda} = 1000\frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n}\nu_i}$

(d)

```
lambda_with_nu = seq(0.5,3, by = 0.001)

constants = lambda_with_nu/1000

middleTerm <- c()
for (i in 1:length(lambda_with_nu)){
  middleTerm[i] <- sum(dark_counts*log(constants[i]*chapter_lengths))
}

log_likelihood_with_nu = -constants*sum(chapter_lengths) + middleTerm - rep(sum(log(factorial(dark_coun

mle_with_nu <- sum(dark_counts)/sum(chapter_lengths) *1000

df_logLikelihood_with_nu = data.frame(x = lambda_with_nu, logLike = log_likelihood_with_nu)
ggplot(df_logLikelihood_with_nu, aes(x = x, y = logLike)) + geom_line(color = 'blue') + geom_vline(xinte
```



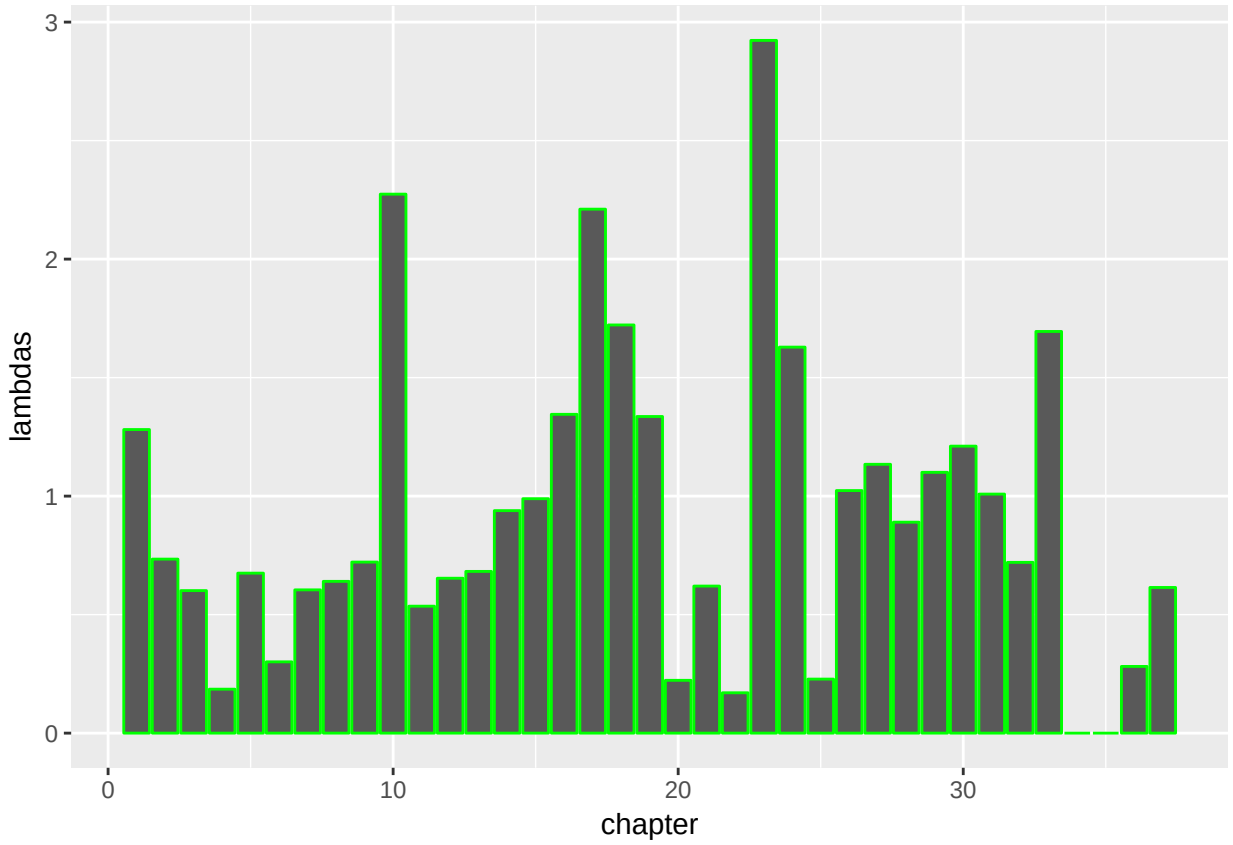
The maximum likelihood estimate, 0.9652801 in this scenario can be interpreted as the best estimate for the

rate at which the word "dark" occurs in each chapter every 1000 words

(e) This is probably not a good assumption since different chapters contain different elements that make up the overall plot.

(f) The MLE for each chapter is similar to that in 2c but in this case since every Y_i is distributed with a different λ the MLE for each chpater is $\hat{\lambda}_i = 1000\frac{y_i}{\lambda_i}$

```
lambda_per_chapter <- 1000 * (dark_counts/chapter_lengths)
lambda_per_chapter <- data.frame(lambdas = lambda_per_chapter, chapter = text_tb$chapter)
ggplot(lambda_per_chapter, aes(x=chapter, y=lambdas)) + geom_bar(stat = 'identity', color = 'green')
```




```
mean(lambda_per_chapter$lambdas)
```

```
## [1] 0.9163078
```

We can also note that the mean of $\lambda_1 \dots \lambda_n$ is equal to 0.9163078.

3

```
chapter_lengths <- word_counts %>% group_by(chapter) %>%
summarize(chapter_length = sum(n)) %>%
ungroup %>% select(chapter_length) %>% unlist %>% as.numeric
```

```
sentiment_map <- get_sentiments("bing")
sentiment_map[sample(nrow(sentiment_map), 10), ]
```

```
## # A tibble: 10 x 2
```

```
##    word           sentiment
##    <chr>          <chr>
##  1 reconciliation positive
##  2 glimmer        positive
##  3 overbalanced   negative
##  4 insurmountable negative
##  5 burned         negative
##  6 degradingly    negative
##  7 suffer         negative
##  8 perversely     negative
##  9 discourteously negative
## 10 extremism      negative
```

```
sentiment_frequencies_mat <- word_counts %>% inner_join(get_sentiments("bing")) %>%
group_by(chapter) %>% summarize(freq_pos = sum(n*(sentiment=="positive"))/sum(n), n=sum(n))
```

```
## Joining, by = "word"
```

```
positive_frequencies <- sentiment_frequencies_mat$freq_pos
```

(a) The following 95% confidence interval is evaluated as follows:

```
p1 <- sentiment_frequencies_mat$freq_pos[17]
n1 <- sentiment_frequencies_mat$n[17]
p2 <- sentiment_frequencies_mat$freq_pos[8]
n2 <- sentiment_frequencies_mat$n[8]

p1
```

```
## [1] 0.3002681
```

```
n1
```

```
## [1] 373
```

```
p2
```

```
## [1] 0.526738
```

```
n2
```

```
## [1] 374
```

```
lowerLimit <- (p1 - p2) - 1.96*sqrt((p1*(1-p1)/n1) + (p2*(1-p2)/n2))
upperLimit <- (p1 - p2) + 1.96*sqrt((p1*(1-p1)/n1) + (p2*(1-p2)/n2))

lowerLimit
```

```
## [1] -0.2952049
```

```
upperLimit
```

```
## [1] -0.1577348
```

By evaluating $(0.3002681 - 0.526738) \pm \sqrt{\frac{0.3002681(1-0.3002681)}{373} + \frac{0.526738(1-0.526738)}{374}}$ we get $(-0.2952049, -0.1577348)$

(b) We are 95% confident that the difference between the positive sentiment scores lies between the $(-0.2952049, -0.1577348)$ meaning chapter 17 has less positive sentiment than chapter 8. Specifically if we took many repeated samples, about 95% of the intervals would contain the true difference in this interval.

(c) Chapter 8 is titled "The Wedding" because Harry is at Bill and Fleur's wedding and he gets to interact with old friends and Ron's entire family, so we expect this chapter to have more positive sentiment than chapter 17 where Harry is reliving his parents death. This agrees with the interval that shows chapter 17 will have significantly less positive sentiment than chapter 8

4

(a)

```
set.seed(123)
lambda_sams <- rgamma(1000, shape=10,scale=1)
mean(lambda_sams)    #emperical mean
```

```
## [1] 9.806898
```

```
var(lambda_sams) #empirical variance
```

```
## [1] 9.316843
```

The empirical mean and empirical variance are almost equal which agrees with the theoretical mean and theoretical variance of a standard Poisson distribution

(b)

	Known	Unknown
Constants	$y_1, \ldots, y_n$, n, α, β	
Variables	$Y_1, \ldots, Y_n$,	λ

(c) $Y_1, \ldots, Y_n$ distributed as $Poi(\lambda_i)$, i.e. $p(y_i) = \frac{\lambda^{y_i} e^{-\lambda}}{y_i!}$ and $\lambda_1, \ldots, \lambda_n$ is distributed as $Gamma(\alpha, \beta)$, i.e. $\pi(\lambda) = \frac{\beta^\alpha}{\Gamma(\alpha)} \lambda^{\alpha-1} e^{-\lambda\beta}$. In the following argument we will use the facts $\Gamma(n) = (n-1)!, \forall n \in \mathbb{N}$ and $\binom{n}{k} = \frac{\Gamma(n+1)}{\Gamma(k+1)\Gamma(n-k+1)}$. Now we compute $p(Y_i \mid \alpha, \beta) = \int p(Y_i \lambda_i \mid \alpha, \beta) d\lambda_i = \frac{\beta^\alpha}{\Gamma(a) y!} \int_0^\infty \lambda^y e^{-\lambda} \lambda^{\alpha-1} e^{-\lambda\beta} = \frac{\beta^\alpha}{\Gamma(a) y!} \int_0^\infty \lambda^{y+\alpha-1} e^{-\lambda(1+\beta)} d\lambda$. Now we have gamma function inside the integral. In order for this integral to converge to one, we need to multiply by the normalizing constant. We need to multiply by the reciprocal of this normalizing constant to remain algebraically true because it is essentially multiplying by 1. Thus we get $\frac{\beta^\alpha}{\Gamma(\alpha) y!} \frac{\Gamma(y+\alpha)}{(1+\beta)^{y+\alpha}} \int_0^\infty \frac{(1+\beta)^{y+\alpha}}{\Gamma(y+\alpha)} \lambda^{y+\alpha-1} e^{-\lambda(1+\beta)} d\lambda$. The inside integral is now a gamma distribution, $Gamma(y+\alpha, 1+\beta)$ and with the normalizing constant this distribution integrates to 1. Leaving us with $\frac{\beta^\alpha}{\Gamma(\alpha) y!} \frac{\Gamma(y+\alpha)}{(1+\beta)^{y+\alpha}}$, by reordering the terms and using the fact $\Gamma(n) = (n-1)!, \forall n \in \mathbb{N}$ we get $\frac{\Gamma(y+\alpha)}{\Gamma(y+1)\Gamma(\alpha)} * \frac{\beta^\alpha}{(1+\beta)^{y+\alpha}}$. Now using the combinotrial and gamma fucntion relation on the first fraction we get $\binom{y+\alpha-1}{y}$ and by factoring the the second fraction we get $(\frac{\beta}{1+\beta})^\alpha (\frac{1}{1+\beta})^y$. Thus we get $\binom{y+\alpha-1}{y} (\frac{\beta}{1+\beta})^\alpha (\frac{1}{1+\beta})^y$ which is a negative binomial distribution with parameters α, $\frac{\beta}{1+\beta}$, i.e. $NB(\alpha, \frac{\beta}{1+\beta})$.

(d) For data generated from a Poisson distribution and the prior distribution of λ (i.e. the paramter of Poisson) coming from a gamma distribution results in a marginal distribution that follows a Gamma. The final mixture results in a negative binomial after factoring and simplifying.