Lab 5: Discrete Random Variables

July 28, 2016

1 Goals

Today we will be viewing tools in `R` for describing and generating random variables

- Derive the mean and variance for the Poisson Distribution
- `R` commands for random variables
- Given a data set, how might we select a model and parameters for the data
- See what happens when the binomial assumptions are broken

2 Deriving the mean and variance of the Poisson Distribution

A derivation can be found here: http://filestore.aqa.org.uk/subjects/AQA-MS03-W-2-SM.PDF

3 R commands for random variables

R has functions for sampling from many different theoretical probability distributions (type '?Distributions' to see the list of available distributions). Every distribution comes with 4 functions, a random draw, a PDF calculation, and 2 calculations for the CDF. They are all prefaced with the letters:

- 'r' - **r**andom draw from the distribution
- 'd' - **d**ensity function (PDF) - returns $P(X = x)$
- 'p' - **p**robability (CDF) - returns $P(X \leq x)$
- 'q' - **q**uantile function (inverse CDF) - returns smallest x such that $P(X \leq x) \geq p$

There is no R function for Bernoulli, but we can use a binomial with $n = 1$. In particular we use "*binom" or "*pois" for the binomial and poisson distributions.

```
# n is number of binomial realizations
# size is the Bernoulli trials
# prob is the probability of success on each Bernoulli trial
rbinom(n = 1, size = 5, prob = .5)

## [1] 2

# pmf of 3, for binomial with n = 5 and p = .5
dbinom(3, size = 5, prob = .5)

## [1] 0.3125
```

```
# cdf of 3, for binomial with n = 5 and p = .5
pbinom(3, size = 5, prob = .5)

## [1] 0.8125
```

3.1 Try it yourself

- Generate a realization of a poisson random variable with mean 7 using `rpois`. In this case, n is the number of realizations and lambda is the mean.
- What is the pmf of 15 for a poisson distribution with mean 10?
- What is the cdf of 6 for a poisson distribution with mean 12?

4 Poisson data

For a poisson distribution, we know that the variance and the mean are both equal to λ. Suppose we have a data set and we want to specify a poisson distribution from which the data might have been generated.

4.1 Questions

- What could I do to select λ?
- How would I assess how well my selected λ reflects my data?

A few possible ways to select the λ parameter are-

- Take the mean of the data
- Take the variance of the data
- Combine the mean and variance of the data somehow

In addition, I could measure the fit of my parameter estimate by calculating how likely the data is to arise from the specified distribution. If the probability is higher, than the parameter is probably a better fit to my data.

Let's first look at how well the mean and the variance estimate the true parameter λ. Just as an example, let's simulate 100 draws form a poisson(10)

```
set.seed(1)
data <- rpois(n = 100, lambda = 10)
mean(data)

## [1] 9.94

var(data)

## [1] 8.400404
```

It looks like the mean is closer to the true lambda than the variance is in this case. Is this always true? Let's repeat this procedure 5000 times and check the results. We also take the log of the probability as a measure of how well the estimates fit our data. In this case, a higher number means a better fitting parameter estimate.

```
set.seed(1)
mean.rec <- rep(0, 5000)
var.rec <- rep(0, 5000)
```

```
ll.m <- ll.v <- rep(0, 5000)

# simulate 5000 data sets and compare the mean and variance's ability to estimate the true lambda
for(i in 1:5000){
  data <- rpois(n = 100, lambda = 10)

  mean.rec[i] <- mean(data)

  var.rec[i] <- var(data)

  ll.m[i] <- sum(log(dpois(data, lambda = mean.rec[i])))
  ll.v[i] <- sum(log(dpois(data, lambda = var.rec[i])))
}

mean(mean.rec)
```

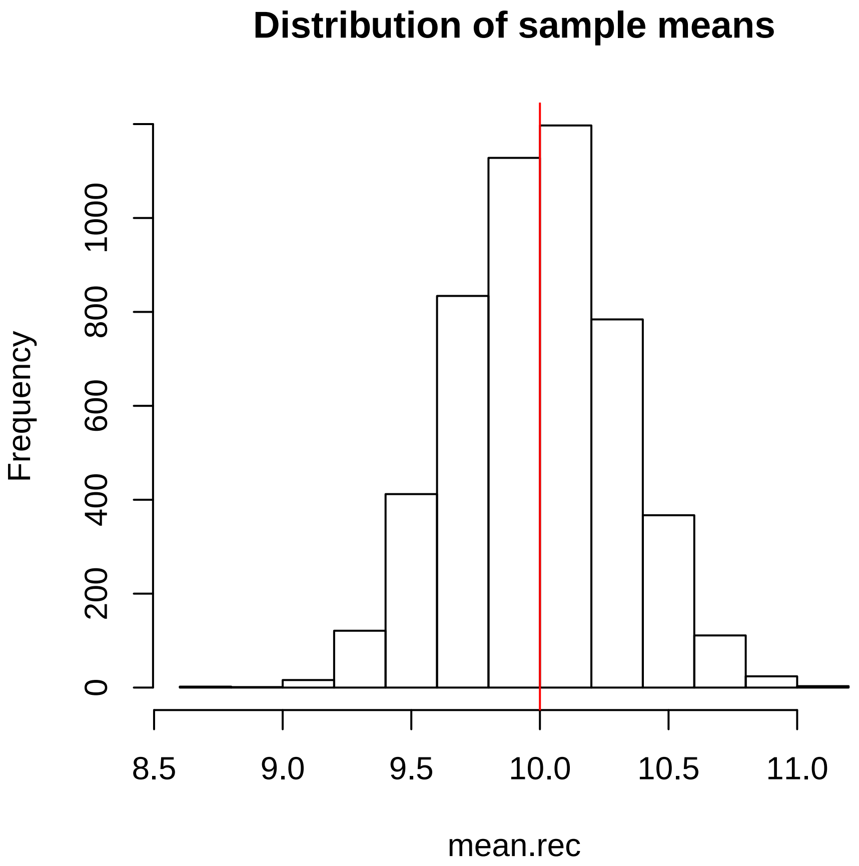
```
## [1] 9.998338
```

```
mean(var.rec)
```

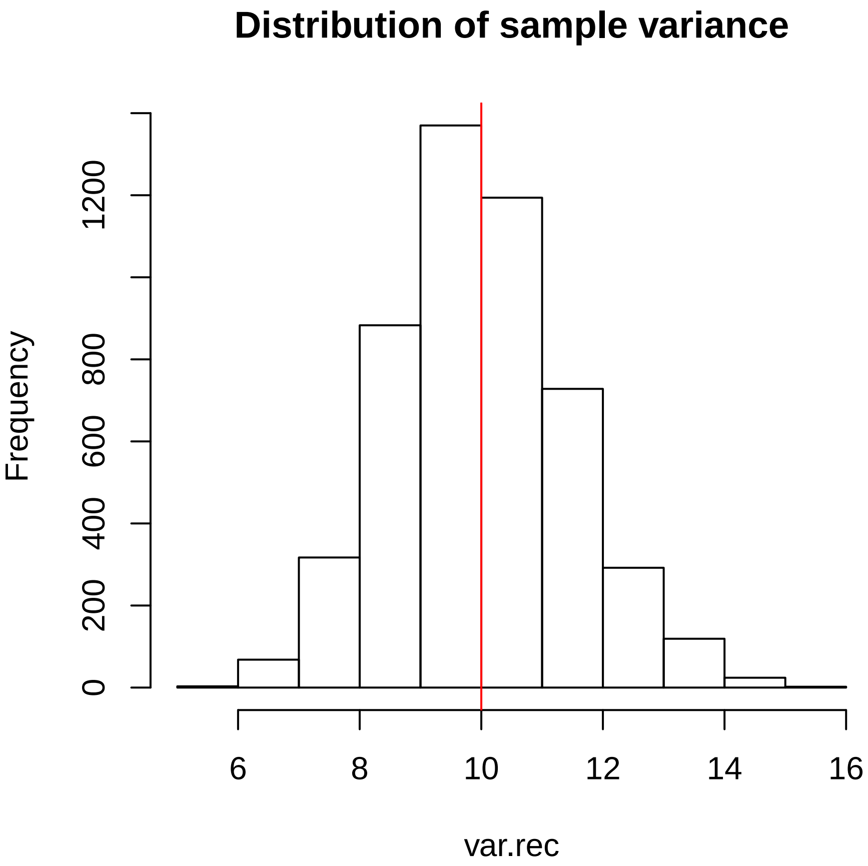
```
## [1] 9.981053
```

We see that the mean of the means and the variances are both almost exactly 10. This confirms that the sample mean is an unbiased estimator for the true mean. It appears that the sample variance is also an unbiased estimator for the true mean. But what about the variability of each of the predictors?

```
hist(mean.rec, main = "Distribution of sample means")
abline(v = 10, col = "red")
```



```
hist(var.rec, main = "Distribution of sample variance")
abline(v = 10, col = "red")
```



Now, let's examine how well the parameter estimates fit our data

```
mean(ll.m)

## [1] -255.556

mean(ll.v)

## [1] -265.7161
```

4.2 Questions

- What can we tell from the histograms above?
- On average, which procedure gives us a better fitting estimate?

5 Try it out yourself

Now repeat the procedure, but with the binomial distribution with $p = .5$ and $n = 10$. We can use the mean of our binomial realizations to directly estimate p, but can we use the variance as well? How would we get an estimate of p from our variance? Is the estimate of p unique?

5.1 Questions

- Repeat the procedure with the binomial distribution
- Do you get similar results to the poisson?
- Do you think this is a general rule, or just specific cases?

6 Breaking Assumptions

The binomial distribution assumes that the Bernoulli trials are independent of each other. What happens if this is not true? Let's form a procedure where the Bernoulli draws are not independent. This type of process might be used to describe birth defects in a litter of cats. For instance, if a kitten has a birth defect, it's also more likely that one of its siblings also has a birth defect.

In the code below, we have a sampling procedure which looks like a binomial random variable. However, after each Bernoulli trial, we update p to be $(.5 + T_1)/(T_2 + 1)$ where T_1 is the number of Bernoulli trials which results in 1 so far, and T_2 is the number of trials so far (regardless of outcome). Thus, if there are a lot of 1's already, we are more likely to have more 1's follow. And if there are a lot of 0's already, we are more likely to have more 0's follow. We let the first Bernoulli have $p = .5$.

Note now instead of defining a variable, we are defining an entire function.

```
dependant.bernoulli <- function(n, size){
  ret <- rep(0, n)

  # for the number of realizations n
  for(i in 1:n){
    t_1 <- 0
    p <- .5
    for(t_2 in 1:size){

      # t_1 keeps track of all the 1's that have been drawn
      t_1 <- t_1 + rbinom(n = 1, size = 1, prob = p)
      # update p
      # note that on the first run, p = .5
      p <- (t_1 + .5) / (t_2 + 1)
    }
    # record the sum of the dependent Bernoulli trials, i.e. one realization of/random draw from our mo
    ret[i] <- t_1
  }

  # return the ret vector
  return(ret)
}
```

Now let's see what happens to the mean and variance of the dependant bernoulli draws.

```
set.seed(2)
draws <- dependant.bernoulli(5000, size = 10)
```

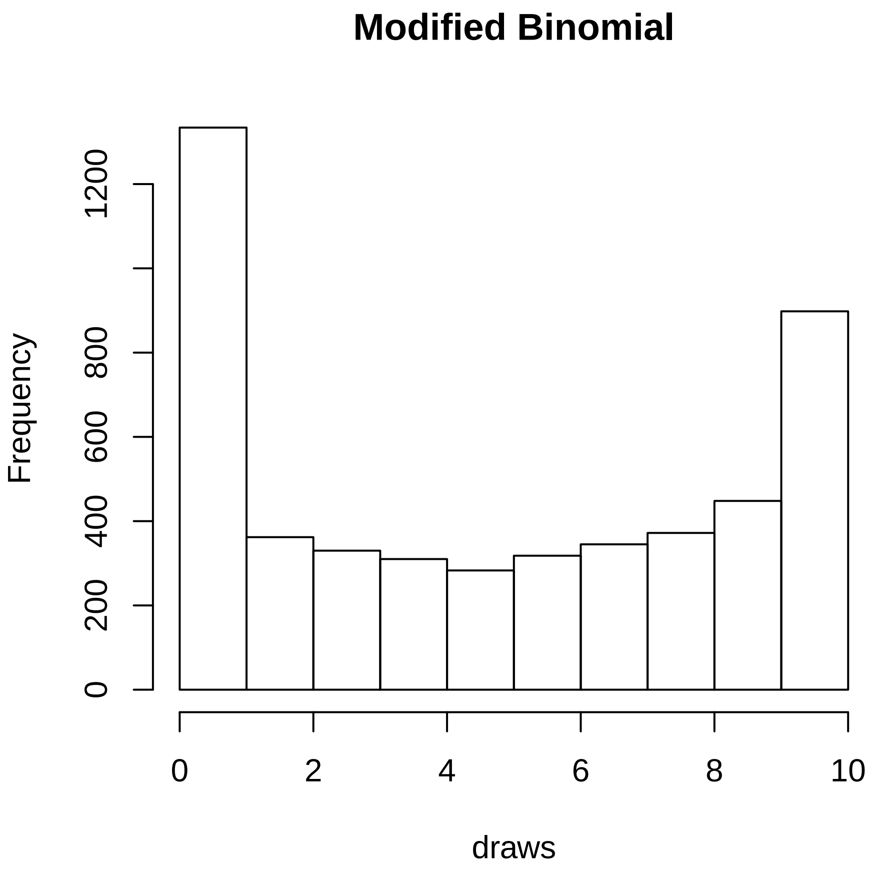
```
mean(draws)

## [1] 5.0308

var(draws)
```

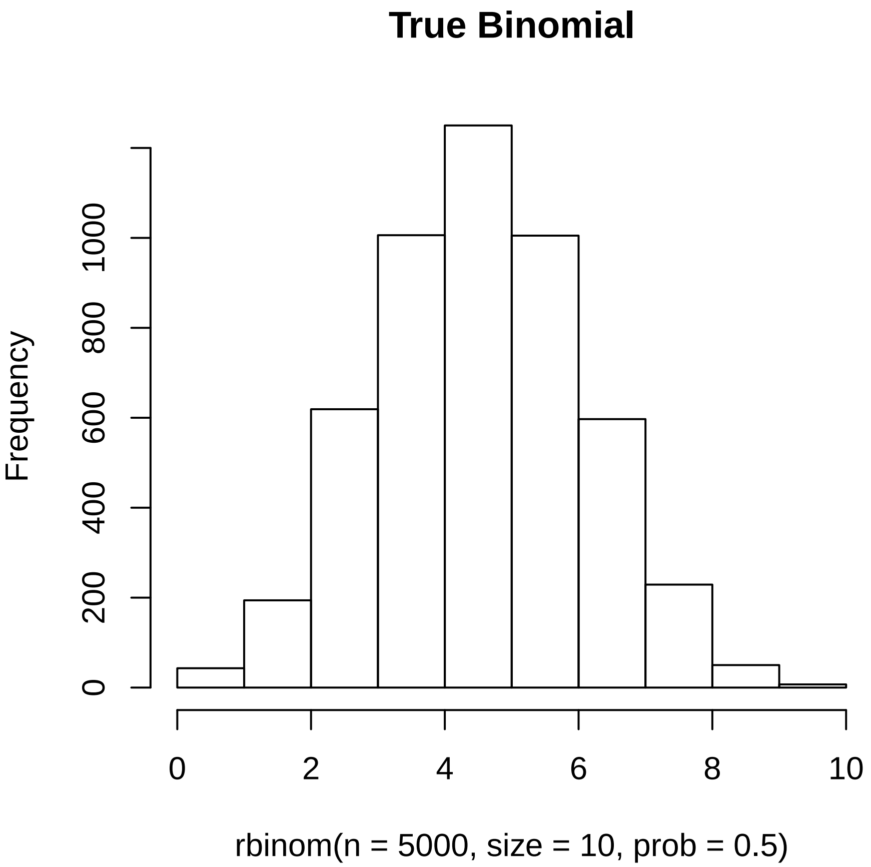
```
## [1] 13.729
```

```r
hist(draws, main = "Modified Binomial")
```



Compare this to what we would see from a normal binomial

```r
hist(rbinom(n = 5000, size = 10, prob = .5), main = "True Binomial")
```



6.1 Questions

- What is the overall probably of each Bernoulli trial now?
- What is the variance of the modified Binomial procedure?
- How does the new mean and variance compare to the normal binomial?
- How do the histograms compare?
- Using concepts about the mean and variance of sums, explain why this happens.