Data Distributions

We're going to review some basic ideas about distributions from Data 8. In addition to review, we introduce some new ideas and emphases to pay attention to:

- Continuous distributions and density curves
- New tools for visualizing and estimating distributions: boxplots and kernel density estimators
- Types of samples and how they effect estimation

1 Basic Exporatory analysis

Let's look at a dataset that you examined briefly in Data 8: flight data from the bureau of transportion with data about the on-time arrival of airplanes in the US. (`http://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=236&DB_Short_Name=On-Time`)

We've streamlined this down to all data concerning departures from SFO in January. Let's explore this data.

```
dataDir <- "../finalDataSets"
flightSF <- read.table(file.path(dataDir, "SFO.txt"),
    sep = "\t", header = TRUE)
dim(flightSF)
```

```
## [1] 13207    64
```

```
names(flightSF)
```

```
##  [1] "Year"                "Quarter"             "Month"
##  [4] "DayofMonth"          "DayOfWeek"           "FlightDate"
```

```
##  [7] "UniqueCarrier"        "AirlineID"            "Carrier"
## [10] "TailNum"              "FlightNum"            "OriginAirportID"
## [13] "OriginAirportSeqID"   "OriginCityMarketID"   "Origin"
## [16] "OriginCityName"       "OriginState"          "OriginStateFips"
## [19] "OriginStateName"      "OriginWac"            "DestAirportID"
## [22] "DestAirportSeqID"     "DestCityMarketID"     "Dest"
## [25] "DestCityName"         "DestState"            "DestStateFips"
## [28] "DestStateName"        "DestWac"              "CRSDepTime"
## [31] "DepTime"              "DepDelay"             "DepDelayMinutes"
## [34] "DepDel15"             "DepartureDelayGroups" "DepTimeBlk"
## [37] "TaxiOut"              "WheelsOff"            "WheelsOn"
## [40] "TaxiIn"               "CRSArrTime"           "ArrTime"
## [43] "ArrDelay"             "ArrDelayMinutes"      "ArrDel15"
## [46] "ArrivalDelayGroups"   "ArrTimeBlk"           "Cancelled"
## [49] "CancellationCode"     "Diverted"             "CRSElapsedTime"
## [52] "ActualElapsedTime"    "AirTime"              "Flights"
## [55] "Distance"             "DistanceGroup"        "CarrierDelay"
## [58] "WeatherDelay"         "NASDelay"             "SecurityDelay"
## [61] "LateAircraftDelay"    "FirstDepTime"         "TotalAddGTime"
## [64] "LongestAddGTime"
```

This dataset contains a lot of information about the flights departing from SFO. We are interested in understanding how frequently flights are delayed (or canceled). Let's look at the column 'DepDelay'. How might we want to explore this data? What single number summaries would make sense? What visualizations could you do?

```
summary(flightSF$DepDelay)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   -25.0    -5.0    -1.0    13.8    12.0   861.0     413
```

Notice the NA's. Let's look at just the subset of some variables for those observations with NA values for departure time (I chosen a few variables so it's easier to look at)

```
naDepDf <- subset(flightSF, is.na(DepDelay))
head(naDepDf[, c("Carrier", "DepDelay", "DepTime",
    "ArrTime", "Cancelled")])
```

```
##     Carrier DepDelay DepTime ArrTime Cancelled
## 44       AA       NA      NA      NA         1
## 75       AA       NA      NA      NA         1
## 112      AA       NA      NA      NA         1
## 138      AA       NA      NA      NA         1
## 139      AA       NA      NA      NA         1
## 140      AA       NA      NA      NA         1
```

```
summary(naDepDf[, c("Carrier", "DepDelay", "DepTime",
    "ArrTime", "Cancelled")])
```

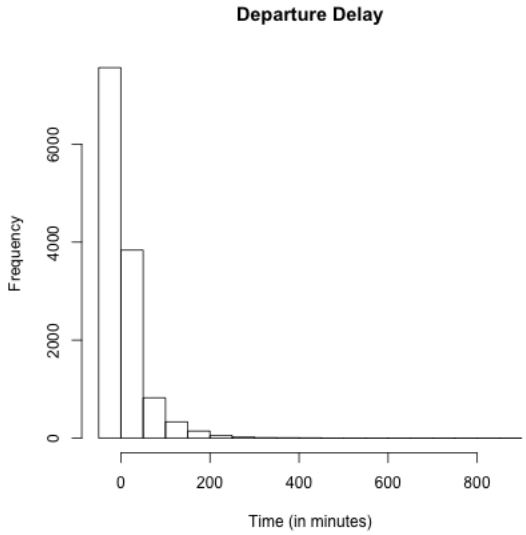
```
##     Carrier       DepDelay      DepTime       ArrTime      Cancelled
##  OO     :176   Min.   : NA   Min.   : NA   Min.   : NA   Min.   :1
##  UA     : 76   1st Qu.: NA   1st Qu.: NA   1st Qu.: NA   1st Qu.:1
##  WN     : 55   Median : NA   Median : NA   Median : NA   Median :1
##  AA     : 35   Mean   :NaN   Mean   :NaN   Mean   :NaN   Mean   :1
##  VX     : 33   3rd Qu.: NA   3rd Qu.: NA   3rd Qu.: NA   3rd Qu.:1
##  DL     : 17   Max.   : NA   Max.   : NA   Max.   : NA   Max.   :1
##  (Other): 21   NA's   :413   NA's   :413   NA's   :413
```

So, the NAs correspond to flights that were cancelled (Cancelled=1).

1.1 Histograms

Let's draw a histogram of the departure delay.

```
hist(flightSF$DepDelay, main = "Departure Delay", xlab = "Time (in minutes)")
```



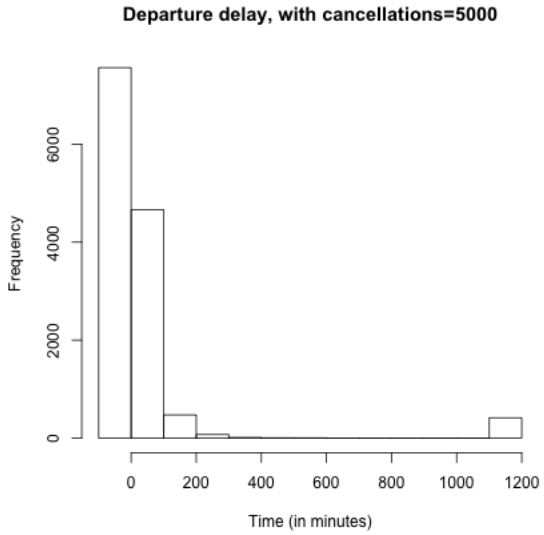
What do you notice about the histogram? What does it tell you about the data?

How good of a summary is the mean or median here? Why are they so different?

Effect of removing data What happen to the NA's? They are just silently not plotted. What does that mean for interpreting the histogram?

We could give the cancelled data a 'fake' value so that it plots.

```
flightSF$DepDelayWithCancel <- flightSF$DepDelay
flightSF$DepDelayWithCancel[is.na(flightSF$DepDelay)] <- 1200
hist(flightSF$DepDelayWithCancel, xlab = "Time (in minutes)",
    main = "Departure delay, with cancellations=5000")
```



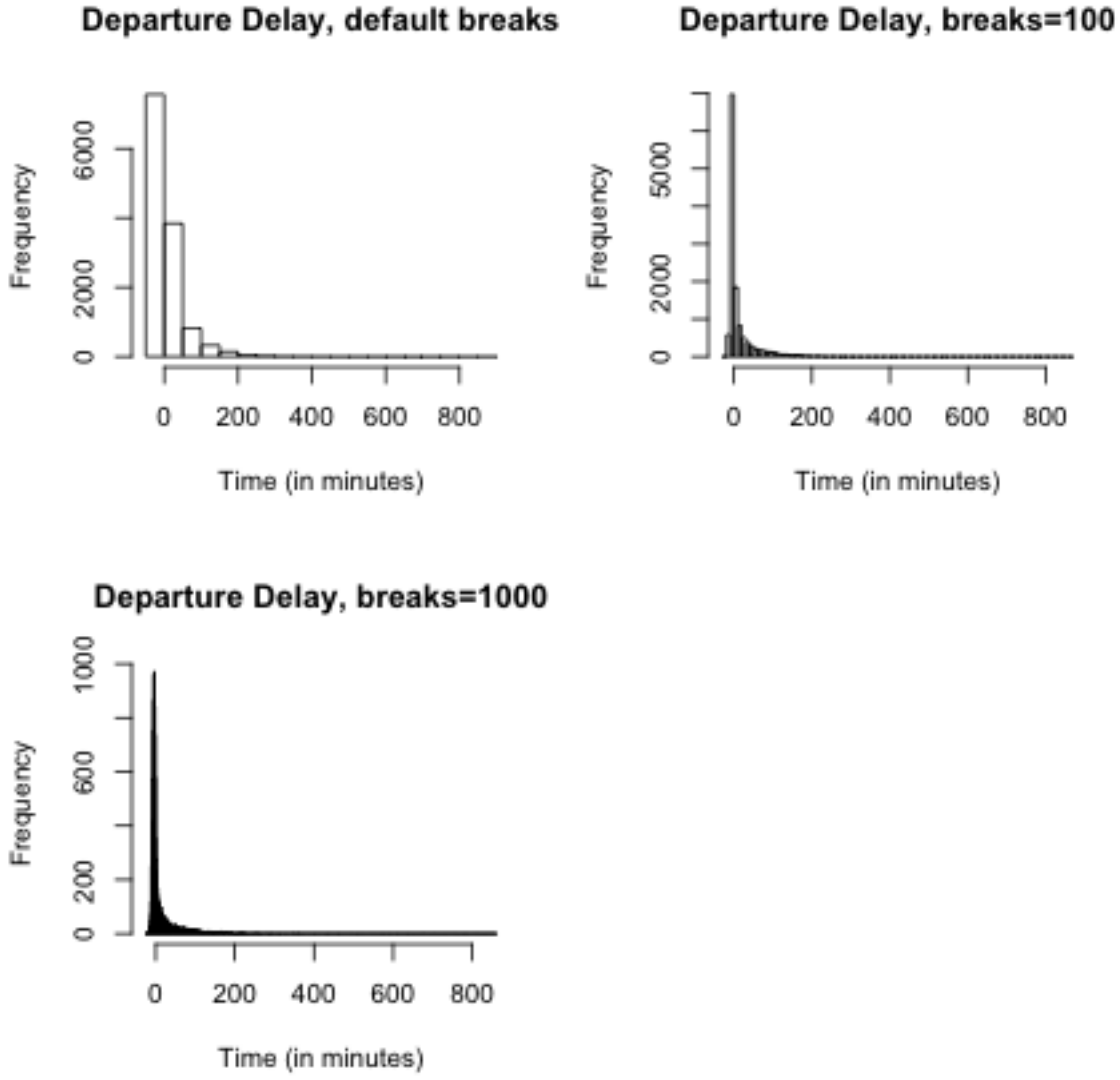
1.1.1 Constructing a Histograms

How do you construct a histogram? Practically, most histograms are created by taking an evenly spaced set of K breaks that span the range of the data, call them $b_1 \leq b_2 \leq ... \leq b_K$ and counting the number of observations in each bin.[1] Then the histogram consists of a series of bars, where the x-coordinates of the rectangles correspond to the range of the bin, and the height corresponds to the number of observations in that bin.

Breaks of Histograms Here's two more histogram of the same data that differ only by the number of breakpoints in making the histograms.

```
par(mfrow = c(2, 2))
hist(flightSF$DepDelay, main = "Departure Delay, default breaks",
    xlab = "Time (in minutes)")
hist(flightSF$DepDelay, main = "Departure Delay, breaks=100",
    xlab = "Time (in minutes)", breaks = 100)
hist(flightSF$DepDelay, main = "Departure Delay, breaks=1000",
    xlab = "Time (in minutes)", breaks = 1000)
```

[1] Recall from data 8 you *can* make a histogram with uneven break points, but this is rather exotic thing to do. If you do, then you have to calculate the height of the bar differently based on the width of the bin because it is the *area* of the bin that should be proportional to the number of entries in a bin, not the height of the bin.



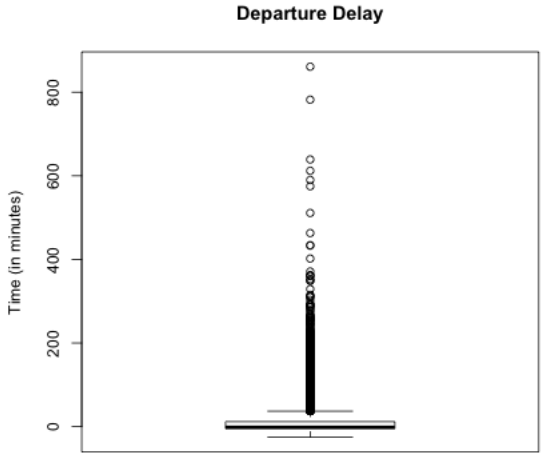
What seems better here? Is there a right number of breaks?

1.2 Boxplots

Another very useful visualization can be a boxplot. A boxplot is like a histogram, in that it gives you a visualization of how the data are distributed. However, its a greater simplification of the distribution. It plots only a box for the bulk of the data, where the limits of the box are the 0.25 and 0.75 quantiles of the data (or 25% and 75% percentiles); a dark line across the middle is the median of the data. In addition, a

boxplot gives additional information to evaluate the extremities of the distribution. It draws 'whiskers' out from the box to indicate how far out is the data beyond the 25% and 75% percentiles; specifically it calculates the interquartitle range (IQR) [just the difference between the 75% and 25% percentiles] and draws the whiskers out 1.5IQR from the boxes – or the smallest/largest data point whichever is closest to the box. Any data points outside of this range are ploted individually.

```
boxplot(flightSF$DepDelay, main = "Departure Delay",
    ylab = "Time (in minutes)")
```



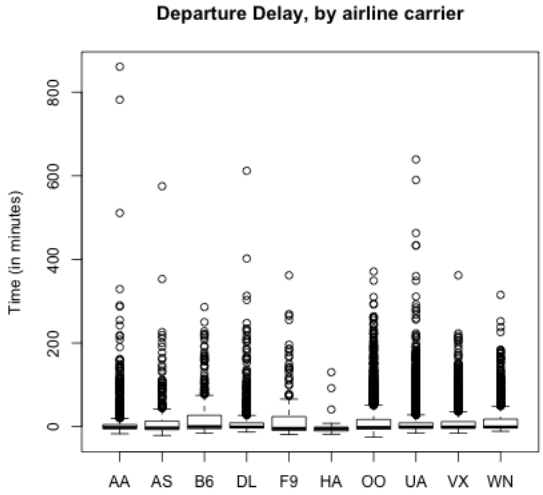
These points are often called "outliers" based on some rules of thumb we won't get into now. However, we can see a lot of data points fall outside this range for our data; this is common for data that is skewed, and doesn't really mean that these points are "wrong", or "unusual" or anything else that we might think about for an outlier.[2]

You might think, why would I want such a limited display of the distribution? First of all, the boxplot emphasizes different things about the distribution. It shows the main parts of the bulk of the data very quickly and simply, and more fine grained information about the extremes ("tails") of the distribution.

Furthermore, because of their simplicity, it is far easier to plot many boxplots and compare them than histograms

[2]If our data had a nice symmetric distribution around the median, like the normal distribution, the rule of thumb would be more appropriate, and this wouldn't happen to the same degree

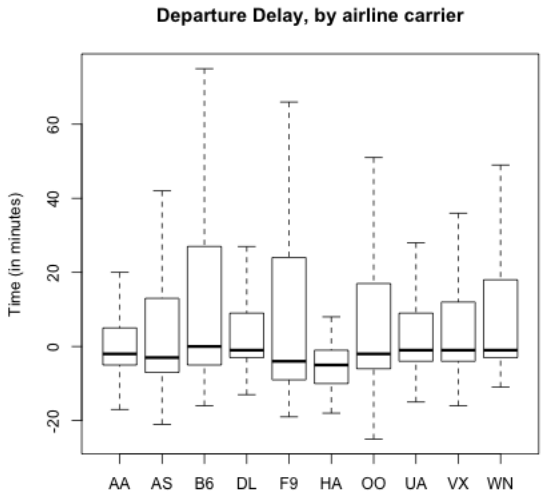
```
boxplot(flightSF$DepDelay ~ flightSF$Carrier, main = "Departure Delay, by airline car
    ylab = "Time (in minutes)")
```



This would be hard to do with histograms.

Notice, I might want to mask all of the "outlier" points as distracting for this comparison,

```
boxplot(flightSF$DepDelay ~ flightSF$Carrier, main = "Departure Delay, by airline car
    ylab = "Time (in minutes)", outline = FALSE)
```



1.3 Descriptive Vocabulary

Here are some useful things to consider in describing distribution of data or comparing two different distributions.

Symmetric refers to equal amounts of data on either side of the 'middle' of the data, i.e. the distribution of the data on one side is the mirror image of the distribution on the other side. This means that the median of the data is roughly equal to the mean.

Skewed refers to when one 'side' of the data spreads out to take on larger values than the other side. More precisely, it refers to where the mean is relative to the median. If the mean $>$ median, then there must be large values on the right-hand side of the distribution, compared to the left hand side (**right skewed**), and if the mean $<$ median then it is the reverse.

Spread refers to how spread out the data is from the middle (e.g. mean or median).

Heavy/light tails refers to how much of the data is concentrated in values far away from the middle, versus close to the middle.

As you can see, several of these terms are mainly relevant for comparing two distributions.[3]

1.4 Transformations

When we have skewed data, it can be difficult to compare the distributions because so much of the data is bunched up on one end, but our axes stretch to cover the large values that are a relatively small proportion of the data. This is also means that our eye focuses on those values too.

A common way to get around this is to transform our data, which simply means we pick a function to transform the data by. For example, a log-transformation of data point y means that we define new data point z so that

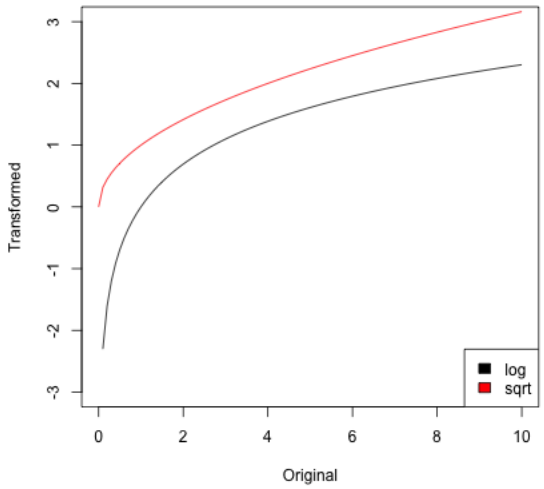
$$z = log(y).$$

A common example of when we want a transformation is for data that are all positive, yet take on values close to zero. In this case, there are often many data points bunched up by zero (because they can't go lower) with a definite right skewed.

[3]But they can often be used without an explicit comparison distribution; in this case, the comparison distribution is always the normal distribution, which is a standard benchmark in statistics

Such data is often nicely spread out for visualization purposes by either the log or square-root transformations.

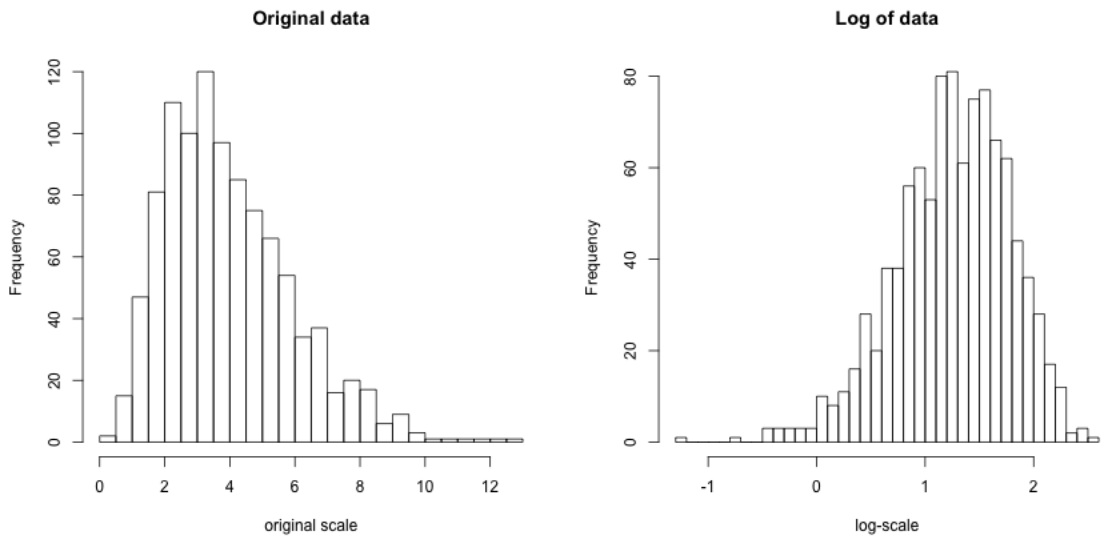
```
ylim <- c(-3, 3)
curve(log, from = 0, to = 10, ylim = ylim, ylab = "Transformed",
    xlab = "Original")
curve(sqrt, from = 0, to = 10, add = TRUE, col = "red")
legend("bottomright", legend = c("log", "sqrt"), fill = c("black",
    "red"))
```



You can see that a two 'data' values of (6,10) that were originally separated by 4, are now separated by roughly 1/2 unit on the transformed scale, while two 'data' values (2,6) that were once also separated by 4, are now separated by roughly 1 unit on the transformed scale. This 'stretches' data on the right to be further apart, while doing the opposite to the left.

I am going to create some fake data with a skew to give an idea of what a transformation does to the data.

```
y <- rgamma(1000, scale = 1, shape = 4)
par(mfrow = c(1, 2))
hist(y, main = "Original data", xlab = "original scale",
    breaks = 30)
hist(log(y), main = "Log of data", xlab = "log-scale",
    breaks = 30)
```

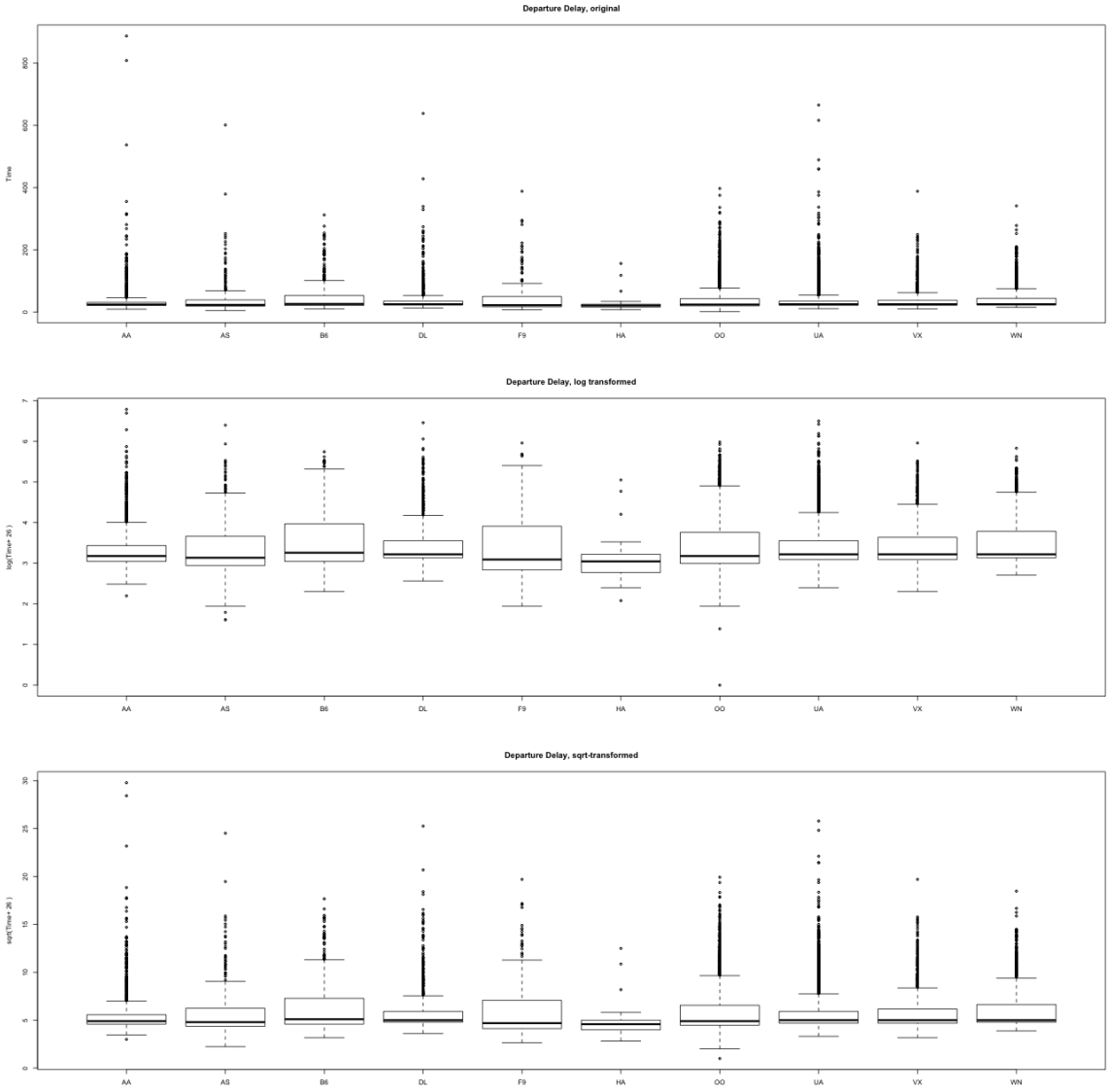


Does it mess up our data? Notice an important property is that these are **monotone** functions, meaning we are preserving the rank of our data – we are not suddenly inverting the relative order of the data. But it does certainly change the meaning when you move to the log-scale. A distance on the log-scale of '2' can imply different distances on the original scale, depending on where the original data was located.[4]

Flight Data Our flight delay data is not so obliging, since it also has negative numbers. But we could, for visualization purposes, shift the data before taking the log or square-root. Here I compare the boxplots of the original data, as well as that of the data after the log and the square-root.

```
addValue <- abs(min(flightSF$DepDelay, na.rm = TRUE)) +
    1
par(mfrow = c(3, 1))
boxplot(flightSF$DepDelay + addValue ~ flightSF$Carrier,
    main = "Departure Delay, original", ylab = "Time")
boxplot(log(flightSF$DepDelay + addValue) ~ flightSF$Carrier,
    main = "Departure Delay, log transformed", ylab = paste("log(Time+",
        addValue, ")"))
boxplot(sqrt(flightSF$DepDelay + addValue) ~ flightSF$Carrier,
    main = "Departure Delay, sqrt-transformed", ylab = paste("sqrt(Time+",
        addValue, ")"))
```

[4]Of course the distance of '2' on the log-scale *does* have a very specific meaning: a distance of '2' on the (base 10) log scale is equivalent to being 100 times greater



Notice that there are fewer 'outliers' and I can see the differences in the bulk of the data better. Did the data become symmetrically distributed or is it still skewed?

2 Discrete Probability Distributions

Let's step back and review basic ideas of sampling and probability distributions that you learned in Data 8.

In the flight data we have *all* flights in the month of January out of SFO. This a

census, i.e. a complete enumeration of the entire population of January flights. You have seen in Data 8 that we can use this data to define the probabilities of events.

Let's assume we want to ask questions about delay times of flights that were not cancelled. For convenience, I'm going to create a new data table that excludes the canceled flights.

```
flightSF_nc <- subset(flightSF, flightSF$Cancelled !=
    1)
```

We could ask, what is the probability that a flight is delayed?

We really need to be more careful, however, because we haven't defined any notion of randomness. If I pick flight AA208 on January 1, 2016 and ask what is the probability it was delayed, this is not a reasonable question, because it either was or wasn't delayed.[5]

So we don't actually want to ask about a particular flight if we are interested in probabilities – we need to have some notion of asking about a randomly selected flight. So let's assume that a flight is randomly selected with all flights having an equal probability of being selected. Now we can ask, what is the probability of such a flight being delayed. Notice that we have exactly defined the randomness mechanism, and so now can calculate probabilities. How would you calculate the following probabilities based on this probability mechanism?

1. $P(\text{flight delay time} = 10 \text{ minutes})$
2. $P(\text{flight delay time} > 2 \text{ hour})$
3. $P(\text{flight is on time})$

This kind of sampling is called a *simple random sample* and is what most people mean when they say "at random." However, there are many other kinds of sampling where not every flight is chosen at random.

Notation We call the delay time of value of a randomly selected flight a *random variable*. We can simplify our notation for probabilities by letting the variable X be short hand for the value of that random variable, and make statements like $P(X > 2)$. We call the complete set of probabilities the *probability distribution* of X.

[5] In fact it wasn't delayed, so you could say the probability was zero that it was delayed.

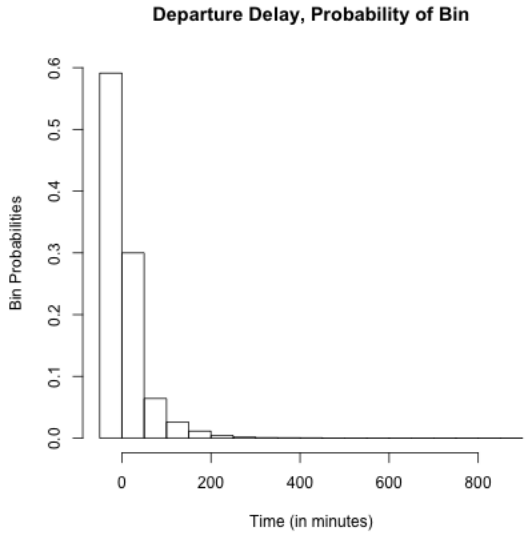
2.1 Probabilities and Histograms

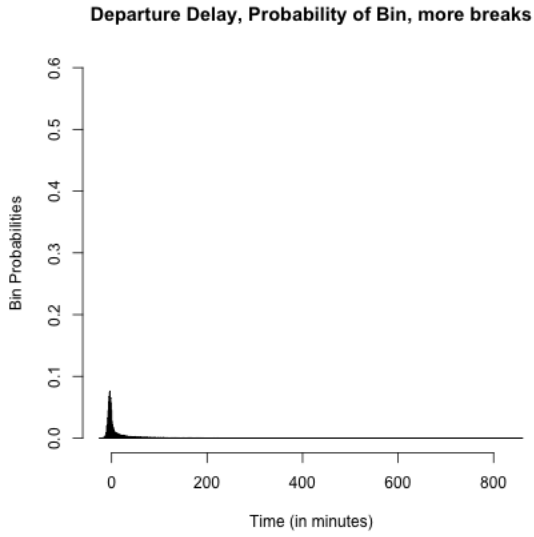
The *frequency histograms* we plotted above give us information about the probabilities of discrete distributions, since they give the count of the numbers of observations in an interval. We can divide that count by the total number of observations, and this gives us the probability of observations lying in each bin.

How would you use the notation above to write this probability, say for the first bin?

Recall, that this is not the same thing as the density of points in a region that you learned about in data 8 – the density of points involves the *area* of a bin. For this reason, plotting the bin probabilities as the height of each bar is NOT what is meant by a histogram.

Plotting these probabilities is not done automatically by R, so we have to manipulate the histogram command in R to do this (and I don't normally recommend that you make this plot – I'm just making it for teaching purposes here). I'll make a little function to do that.





2.2 What about those cancelled flights? (Conditioning)

We asked a question of the population of non-cancelled flights, so that X is the random variable corresponding to delay times of a randomly selected flight from *that*

population.

Suppose instead I want to also include cancelled flights in my population of flights, i.e. all flights. To avoid confusion we can call this random variable Y. Y is equal to the delay time if the flight isn't cancelled, and otherwise we'll say its equal to NA (like in our data encoding). How could I calculate the probability being on time from a randomly selected flight from this population ($P(Y <= 0)$)?

What about probability of being delayed ($P(Y > 0)$)?

Is the probability distribution of Y the same as the probability distribution of X?

There is a relationship between the random variables X and Y. If you continually generated Y and only keep those realizations of Y that aren't a cancelled flight, then you would get data that has the same probability distribution as X. The random data that we get out by only keeping some data is a random variable, sometimes written as $Y|Y \neq \text{NA}$. Then the probability distribution of $Y|Y \neq \text{NA}$ takes on a is called a *conditional probability distribution*. We can write

$$P(Y = 10|Y \neq \text{NA})$$

which of course, is the same as $P(X = 10)$. This is true for all possible probability statements about $Y|Y \neq \text{NA}$, meaning that $Y|Y \neq \text{NA} = X$ – they define the same random variable/probability distribution as we said before. So basically, instead of defining a separate random variable for each possible conditioning, we can use the conditioning notation. This has the benefit of making sure that we always remember that we are ignoring those cancelled flights.

3 Histograms of samples of data

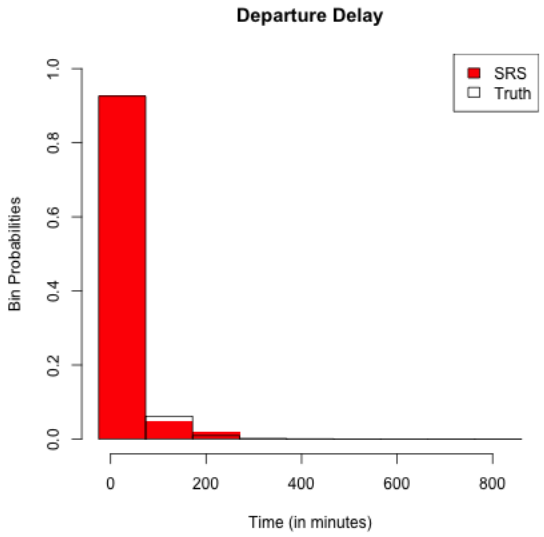
Generally the data we work with is a sample, not the complete population.

Consider what happens if you take a simple random sample of 100 flights from our complete set of flights and calculate a histogram. For simplicity we will sample from the population of flights in January *without* a cancellation.

```
flightSRS <- sample(x = flightSF_nc$DepDelay, size = 100,
    replace = TRUE)
```

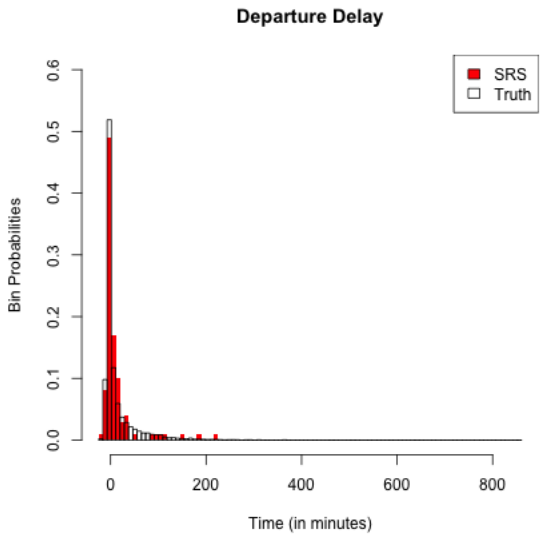
Let's draw a plot giving the proportions of the total sample in each bin (i.e. not a histogram). I'm going to also draw the true population probabilities of being in each bin as well, and put it on the same histogram as the sample proportions. To make sure they are using the same breakpoints, I'm going to define the break points manually. (Otherwise the specific breakpoints will depend on the range of each dataset and so be different)

```
ylim <- c(0, 1)
breaks <- seq(min(flightSF_nc$DepDelay), max(flightSF_nc$DepDelay),
    length = 10)
histBinProb(flightSRS, main = "Departure Delay", xlab = "Time (in minutes)",
    border = NA, breaks = breaks, ylim = ylim, col = "red",
    add = FALSE)
histBinProb(flightSF_nc$DepDelay, main = "Departure Delay",
    xlab = "Time (in minutes)", col = NULL, border = "black",
    breaks = breaks, ylim = ylim, lwd = 2, add = TRUE)
legend("topright", c("SRS", "Truth"), fill = c("red",
    "white"))
```



Pretty good. Suppose I had smaller width breakpoints (next figure), what conclusions would you make?

```
ylim <- c(0, 0.6)
breaks <- seq(min(flightSF_nc$DepDelay), max(flightSF_nc$DepDelay),
    length = 100)
histBinProb(flightSRS, main = "Departure Delay", xlab = "Time (in minutes)",
    border = NA, breaks = breaks, ylim = ylim, col = "red",
    add = FALSE)
histBinProb(flightSF_nc$DepDelay, main = "Departure Delay",
    xlab = "Time (in minutes)", col = NULL, border = "black",
    breaks = breaks, ylim = ylim, lwd = 2, add = TRUE,
    lwd = 3)
legend("topright", c("SRS", "Truth"), fill = c("red",
    "white"))
```



Histograms as Estimates So when we are working with a sample of data, we should always think of probabilities obtained from a sample as an *estimate* of the probabilities of the full population distribution. This means histograms, boxplots, quantiles, and any estimate of the probability have variability, like any other estimate.

This means we need to be careful about the dual use of histograms as both visualization tools and estimates. As visualization tools, they are always appropriate for understanding *the data you have*: whether it is skewed, whether there are outlying or strange points, etc.

To draw broader conclusions from histograms or boxplots performed on a sample, however, is to view them as estimates of the entire population. Then you need to think carefully about how the data was collected.

Different Types of Samples For example, let's consider that I want to understand if the distribution of delay times for United flights is similar to that of American in 2015/2016 academic year. Consider the following *samples* of data

- All flights in January
- A simple random sample drawn from all flights in the 2015/2016 academic year.
- 12 separate simple random samples drawn from every month in the 2015/2016 academic year, combined together into a single dataset

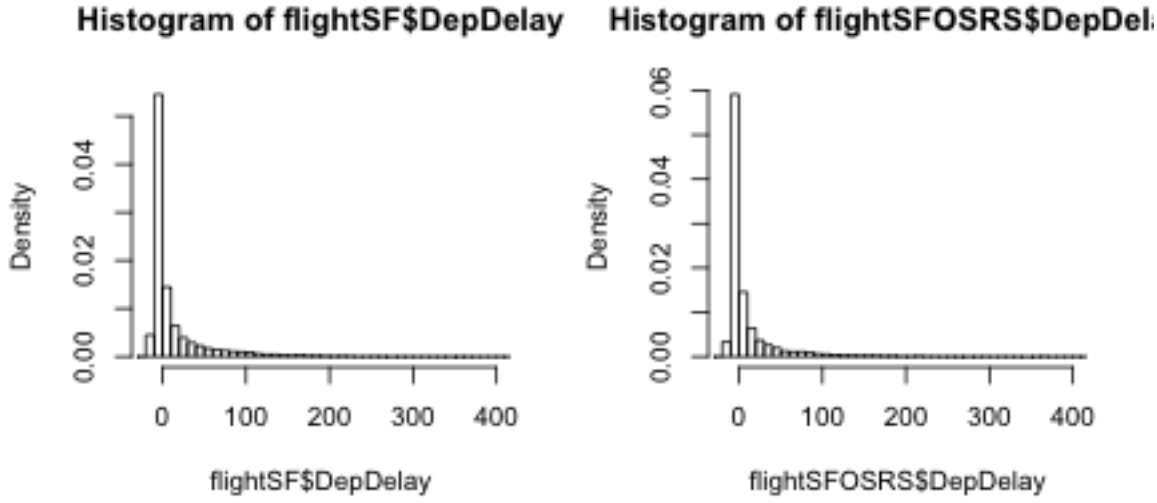
Why do I now consider all flights in January as a sample, when before I said it was a census?

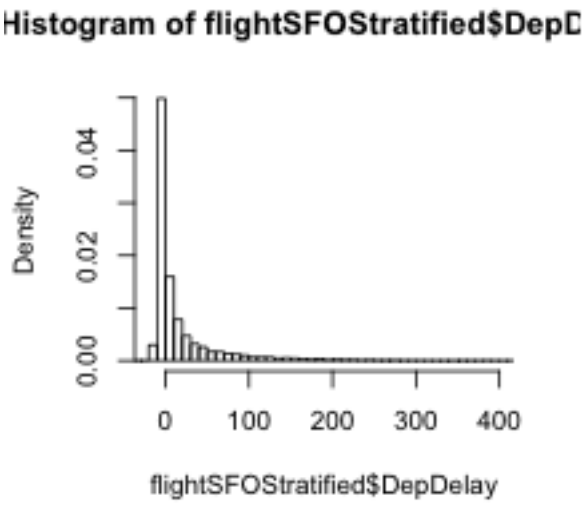
All three of these are samples from the population of interest and we can assume that we choose to make them have the same sample size.

One is not a *random sample* (which one?). Only one is a *simple random sample*. The last sampling scheme, created by doing a SRS of each month and combining the results, is also a random sampling scheme, but not a SRS (it is called a *Stratified random sample*). We know it's random because if we did it again, we wouldn't get exactly the same set of data (unlike our January data).

If we draw histograms of these different samples, they will all describe the distribution of the sample, but they will not all be good estimates of the underlying population distribution. We have access to all of the flight data from the months, so we can actually make both of these datasets.

```
flightSFOSRS <- read.table(file.path(dataDir, "SFO_SRS.txt"),
    sep = "\t", header = TRUE, stringsAsFactors = FALSE)
flightSFOStratified <- read.table(file.path(dataDir,
    "SFO_Stratified.txt"), sep = "\t", header = TRUE,
    stringsAsFactors = FALSE)
par(mfrow = c(2, 2))
xlim <- c(-20, 400)
hist(flightSF$DepDelay, breaks = 100, xlim = xlim,
    freq = FALSE)
hist(flightSFOSRS$DepDelay, breaks = 100, xlim = xlim,
    freq = FALSE)
hist(flightSFOStratified$DepDelay, breaks = 100, xlim = xlim,
    freq = FALSE)
```





How do these histograms compare?

In particular, drawing histograms or estimating probabilities as you have learned in Data 8 only give good estimates of the population distribution *if the data is a SRS.* Otherwise they can vary quite dramatically from the actual population.

So are only SRS good random samples? NO! The stratified random sample described above can actually be a much better way to get a random sample and give you *better* estimates – but you must correctly create your estimates. For the case of the histogram, you have to estimate the histogram in such a way that it

correctly estimates the distribution of population, rather than the distribution of the sample. How? The key thing is that because it is a random sample, drawn according to a *known probability mechanism*, it is possible to make a correct estimate of the population.

How to make these kind of estimates for random samples that are not SRS is beyond the scope of this class, but there are standard ways to do so for stratified samples and many other sampling designs. Indeed most national surveys, particularly the high-quality ones produced by the national government, are not SRS but much more complicated sampling schemes that can give equally accurate estimates, but often with less cost.

4 Continuous Distributions

Data 8 primarily relied on **discrete distributions**, meaning that the possible values that can be observed is a finite set of values. For example, if we draw a random sample from our flight data we know that only the 289 unique values of the flights in January can be observed – not all numeric values are possible (no decimals are possible, for example). Not even all possible integers in a range are seen (you could get a delay time of 250 minutes, but not 251 minutes).

When you do hypothesis testing with permutation tests and bootstrap methods, you are also (re)sampling from a discrete distribution – the set of data points observed.

However, it can be useful to think about probability distributions that allow for all numeric values (i.e. continuous values), *even when we know the actual population is finite*. These are **continuous distributions**.

For example, suppose we wanted to use this set of flights in January to decide which airlines we should use in the future. It's more reasonable to think that there is an (unknown) probability distribution that defines what we expect to see for that data that is defined on a continuous range of values.

Of course some features of the data are "naturally" discrete, like the set of airline carriers, and there no rational way to think of them being continuous.

4.1 Probability with Continuous distributions

Some probability ideas become more complicated/nuanced for continuous distributions. In particular, for a discrete distribution, it makes sense to say $P(X = 10)$

(the probability of a 10 minute flight delay). For continuous distributions, such an innocent statement is actually fraught with problems.

To see why, remember what you know about discrete probability distributions. In particular,

$$0 \leq P(X = 10) \leq 1$$

Furthermore, any probability statment has to have this property, not just ones involving '=': e.g. $P(X \leq 10)$ or $P(X \geq 0)$. This is a fundamental rule of probability, and also holds true for continuous distributions.

Okay so far. Now another thing you learned is if I give all possible values that my random variable X can take (the *sample space*) and call them $v_1, \ldots, v_K$, then if I sum up all these probabilities they must sum exactly to 1,

$$\sum_{i=1}^{K} P(X = v_i) = 1$$

Furthermore, $P(X \in \{v_1, \ldots, v_K\}) = 1$, i.e. the probability X is in the sample space must of course be 1.

Well this becomes more complicated for continuous values – this leads us to an infinite sum since we have an infinite number of possible values. Moreover, if we have any positive probability (i.e. $\neq 0$) for each point in the sample space, then we won't 'sum' to one [6] These kinds of concepts from discrete probability just don't translate over exactly to continuous.

To deal with this, *continuous distributions do not allow any positive probability for a single value*: if X has a continuous distribution, then $P(X = x) = 0$ for any value of x. Instead, continuous distributions only allow for positive probability of an interval: $P(x_1 \leq X \leq x_2)$ can be greater than 0.

This isn't so strange if you think about it. What is your intuitive sense of the probability of a flight delay of exactly 10 minutes – and not 10 minutes 10 sec or 9 minutes 45 sec? You see that once you allow for this kind of precision, it is actually reasonable to say that exactly 10 minutes has no real probability that you need worry about.

What if you want the chance of getting a 10 minute flight delay? Well, you really mean a small interval around 10 minutes, since there's a limit to our measurement ability anyway. This is what we also do with continuous distributions: we discuss the

[6] For those with more math: convergent infinite series can of course sum to 1. But we are working with the continuous real line (or an interval of the real line), and there is not bijection between the integers and the continuous line.

probability in terms of increasingly small intervals around 10 minutes. The mathemtatics of calculus give us the tools to do this[7], but we are not going to actually use those calculus tools in this class. For any continuous distributions you can think of, the computer will give us these results. Instead we are going to focus on the big ideas, which is all that is needed.

4.2 Probability Density Functions (pdfs)

For discrete distributions, we can completely describe the distribution of a random variable by describing the probability of each of the discrete values it takes on. Knowing $P(X = v_i)$ for all possible values of v_i in the sample space completely defines the probability distribution.

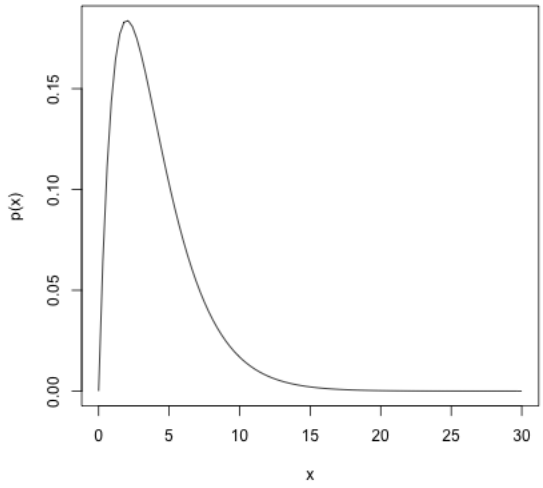
If we can't talk about $P(X = x)$, then how do we define a continous distribution? Instead we talk about a **probability density function (pdf)**. A probability density function is a function $p(x)$, so that if you draw this function and measure the area under its curve for an interval, it gives you probability of that interval. So let's break that down.

Let's take the following pdf, which is perhaps vaguely similar to our flight data, though on a different scale

$$p(x) = \frac{1}{4} x e^{-x/2}$$

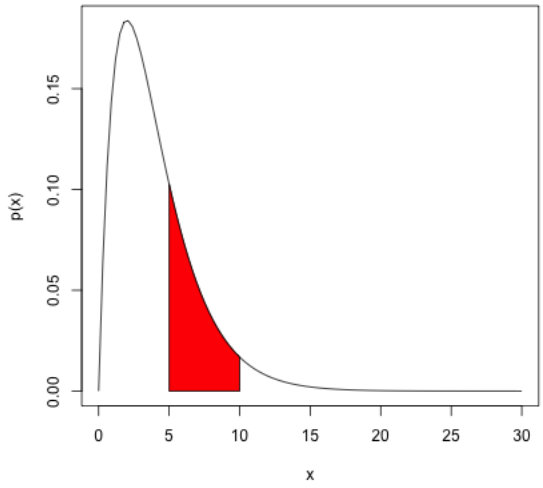
```
curve(x * exp(-x/2)/4, xlim = c(0, 30), ylab = "p(x)",
    xlab = "x")
```

[7] If you've taken calculus, you probably recall the idea of integration giving the area under a curve, or derivatives as being the rate of change in $f(x)$ for an infintesimally small amount of change in x



Suppose that X is a random variable from a distribution with this pdf. Then to find $P(5 \leq X \leq 10)$, I find "the area under the curve" of $p(x)$ between 5 and 10.

```
plotUnderCurve <- function(x1, x2, p, ...) {
    x = seq(x1, x2, len = 100)
    y = p(x)
    polygon(c(x, tail(x, 1), x[1]), c(y, 0, 0), ...)
}
p <- function(x) {
    x * exp(-x/2)/4
}
curve(p, xlim = c(0, 30), ylab = "p(x)", xlab = "x")
plotUnderCurve(5, 10, p, col = "red")
```



Our same rule from discrete distribution applies, namely that the probability of X being in the entire sample space must be 1. What does this mean in terms of the cumulative area under the curve of $p(x)$?

Key properties of continuous distributions (for this class at least!)

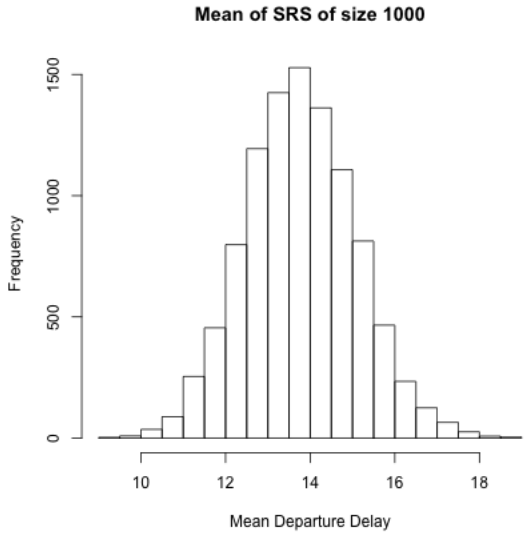
1. Probabilities are always between 0 and 1, inclusive.
2. Probabilities are only calculated for intervals, not individual points
3. A probability density function (pdf) defines a continuous distribution and gives the probability of any interval by taking the area under the curve

4.2.1 Normal Distribution and Central Limit Theorem

You've seen a continuous distribution when you learned about the central limit theorem in Data 8.

Recall, if I take a SRS of a population and calculate it's mean, say $\bar{X}$, this is a random variable that has a distribution. It's randomness is due to the randomness in the SRS. If I do this many times I can look at the distribution of $\bar{X}$

```
sampleSize <- 1000
sampleMean <- replicate(n = 10000, expr = mean(sample(flightSF_nc$DepDelay,
    size = sampleSize, replace = TRUE)))
hist(sampleMean, xlab = "Mean Departure Delay", main = paste("Mean of SRS of size",
    sampleSize))
```



If the size of the sample is large enough, the distribution (i.e. histogram) of $\bar{X}$ will look like a bell shaped curve. The central limit theorem tells that for large sample sizes, this always happens, *regardless of the original distribution of the data.* This curve is called the *normal distribution.*

A normal distribution has two **parameters** that define the distribution, its mean μ and variance σ^2. It's pdf is

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{\sigma^2}}$$

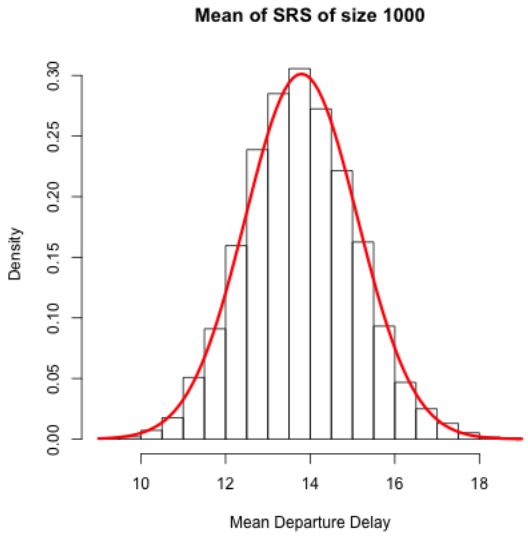
It's a mouthful, but easy for a computer to evaluate.

Then the central limit theorem says that if the original distribution has mean μ_{true} and variance τ^2_{true}, then the distribution of $\bar{X}$ for a sample of size n will be approximately

$$N(\mu_{true}, \frac{\tau^2_{true}}{n})$$

Back to Flight data We can overlay the normal distribution on our histogram, if we draw a density histogram (i.e. scale the frequencies so that the area under the

curve sums to 1). Notice we also have to pick the right mean and standard deviation for our normal distribution for these to align. For most actual datasets, of course, we don't know the true mean of the population, but since we sampled from a known population we do.



Probabilities of a normal distribution Recall that for a normal distribution, the probability of being with in 1 standard deviation of μ is roughly 0.68 and the probability of being within 2 standard deviations of μ is roughly 0.95.

What is the probability that a observed random variable from a $N(\mu, \sigma^2)$ distribution is *less* than μ by more than 2σ?

For $\bar{X}$, which is approximately normal, if the original population had mean μ and variance τ, the standard deviation of that normal is $\tau/\sqrt{n}$. What does this mean for the chance of a single mean calculated from your data being far from the true mean (relate your answer to the above information about probabilities in a normal)?

It also means that trying to improve your estimate by getting 100 observations, will result in different improvements in your estimate depending on your sample size. For example, if you had $n = 1000$ and you go to $n = 1100$, that will make your mean likely to be within $\tau/16.5$ instead of $\tau/15$. In comparison, if you only have $n = 20$ observations, getting 100 additional observations will change from being likely within $\tau/2.2$ of the true mean to being within $\tau/5.5$, which is a much bigger deduction in

the range of likely values for your mean.

4.2.2 More on density curves

"Not much good to me" you might think – you can't evaluate $p(x)$ and get any probabilities out. It just requires the new task of finding an area. However, finding areas under curves is routine with calculus tools (it's called integration), and even if there is not a analytical solution, the computer can calculate the area. So pdfs are actually quite useful.

Moreover, $p(x)$ *is* interpretable, just not as a direct tool for probability calculations. For smaller and smaller intervals you are getting close to the idea of the "probability" of $X = 10$. For this reason, where discrete distributions use $P(X = 10)$, the closest corresponding idea for continuous distributions is $p(10)$: though $p(10)$ is not a probability like $P(X = 10)$ the value of $p(x)$ gives you an idea of more likely regions of data.

More intuitively, the curve $p(x)$ corresponds to the idea of of a histogram of data. It's shape tells you about where the data are likely to be found, just like the bins of the histogram. We see for our example of $\bar{X}$ that the histogram of $\bar{X}$ (when properly plotted on a density scale) approaches the smooth curve of a normal distribution. So the same intuition we have from the discrete histograms carry over to pdfs.

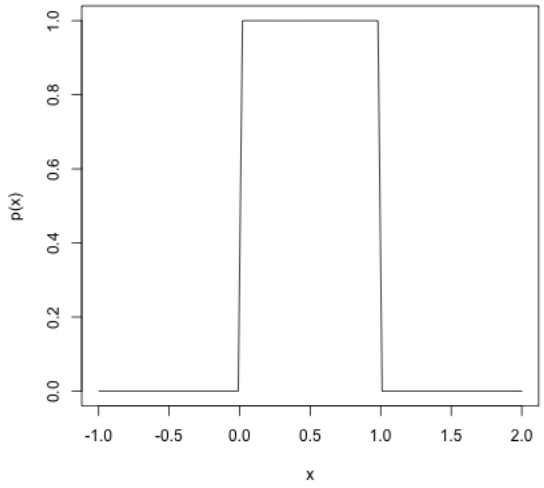
Properties of pdfs

1. The total area under the curve $p(x)$ must be exactly equal to 1
2. Unlike probabilities, the value of $p(x)$ can be ≥ 1 (!).

This last one is surprising to people, but $p(x)$ is not a probability – only the area under it's curve.

To understand this, consider this very simple density function:

$$p(x) = \begin{cases} 1 & x \in [0, 1] \\ 0 & x > 1, x < 0 \end{cases}$$

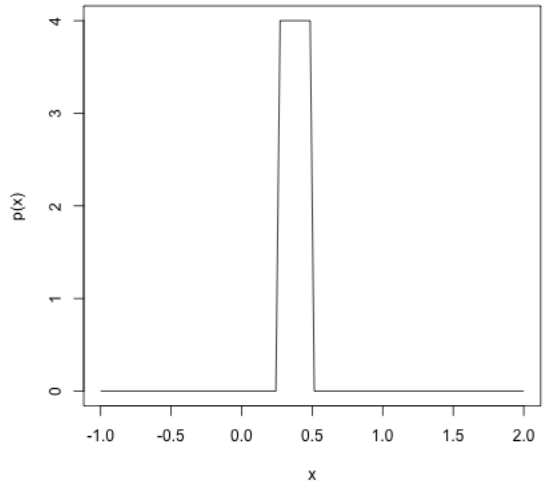


This is a density function that corresponds to being equally likely for any value between 0 and 1; why?

What is the area under this curve? Well it's just a rectangle, so...

it's called a *uniform distribution* on [0,1], some times abbreviated $U(0,1)$.

Suppose instead, I want density function that corresponds to being equally likely for any value between 1/4 and 1/2 (i.e. $U(1/4,1/2)$).



Then again, we can easily calculate this area . If $p(x)$ was required to be less than one, you couldn't get the total area to be 1.

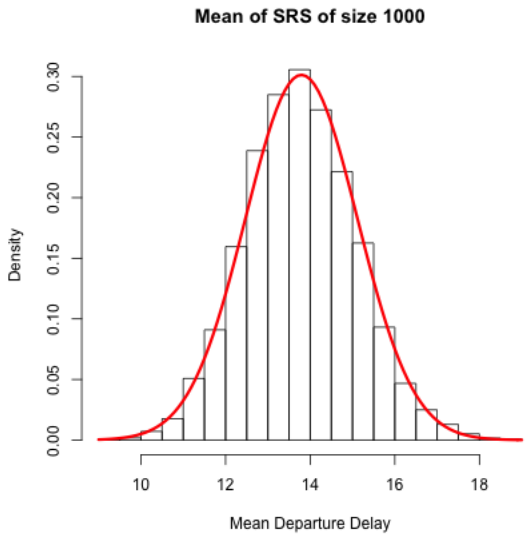
So you see that the scale of values that X takes on matters to the value of $p(x)$. If X is concentrated on a small interval, then the density function will be quite large, while if it is diffuse over a large area the value of the density function will be small.

Example: Changing the scale of measurements : Suppose my random variable X are measurements in centimeters, with a normal distribution, $N(\mu = 100\text{cm}, \sigma^2 = 100\text{cm}^2)$. What is the standard deviation?

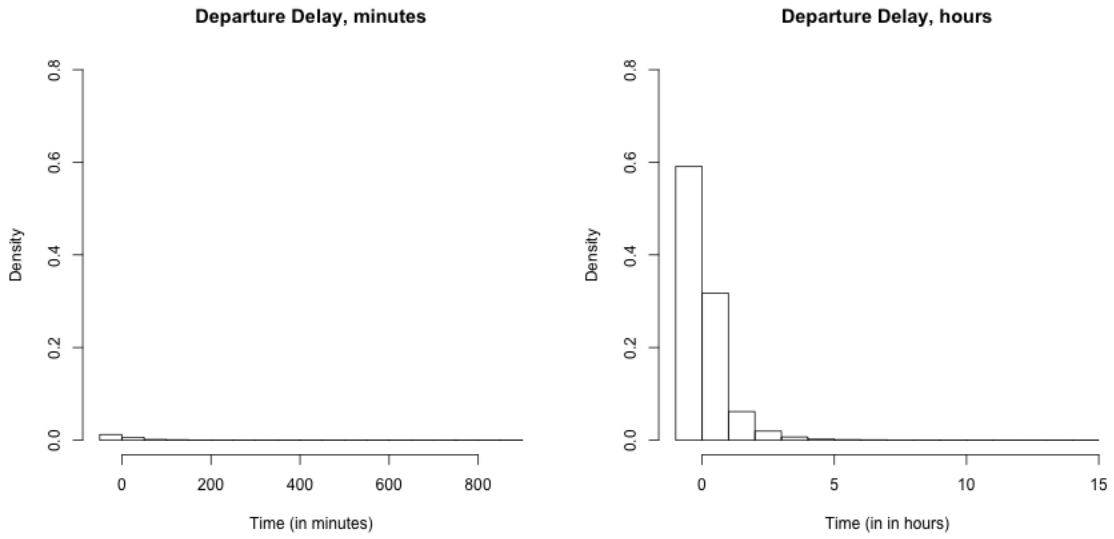
Then I decide to convert all the measurements to meters (FYI: 100 centimeters=1 meter). What is now the mean? And standard deviation?

Density Histograms We've been showing histograms with the frequency of counts in each bin on the y-axis. But , histograms are meant to represent the distribution of continuous measurements, so they are defined to approximate density functions. Specifically, histograms are properly drawn on the density scale, meaning that you want the total area in all of the rectangles of the histogram to have area one. Notice how when I overlay the normal curve for discussing the central limit theorem, I had to set my `hist` function to `freq=FALSE` to get proper density histograms. Otherwise the histogram is on the wrong scale.

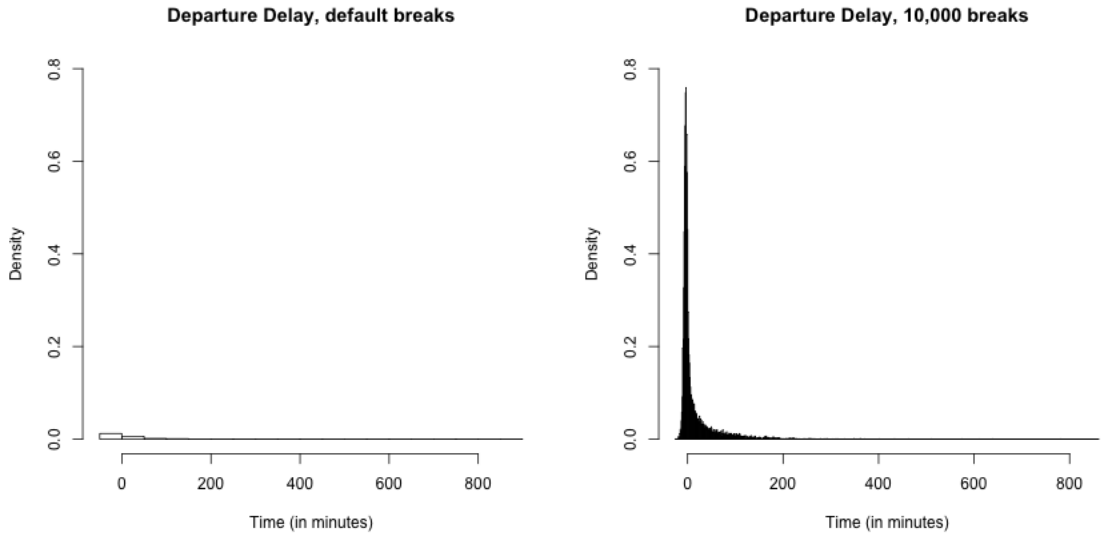
```
hist(sampleMean, xlab = "Mean Departure Delay", main = paste("Mean of SRS of size",
    sampleSize), freq = FALSE)
m <- mean(flightSF_nc$DepDelay)
s <- sqrt(var(flightSF_nc$DepDelay)/sampleSize)
p <- function(x) {
    dnorm(x, mean = m, sd = s)
}
curve(p, add = TRUE, col = "red", lwd = 3)
```



Therefore, just like density curves, if you plot histograms on the density scale, you can get values greater than 1.

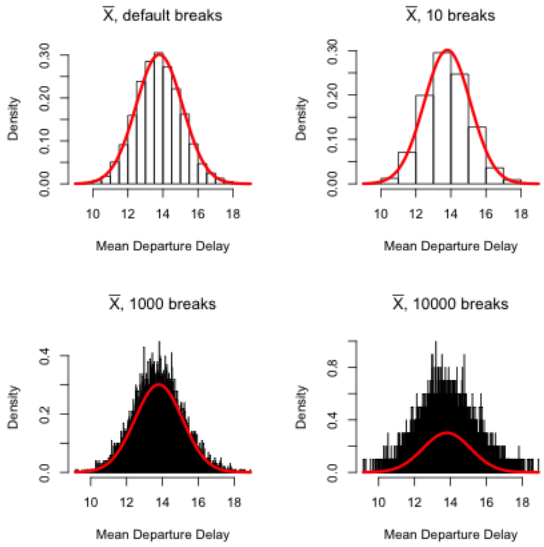


Notice how density values vary (like counts) as you change the breaks. Why?



So you can see how the breaks in a histogram can affect their ability to be a good *estimate* of the density. Consider our sample of $\bar{X}$ values, which we know approximates a normal,

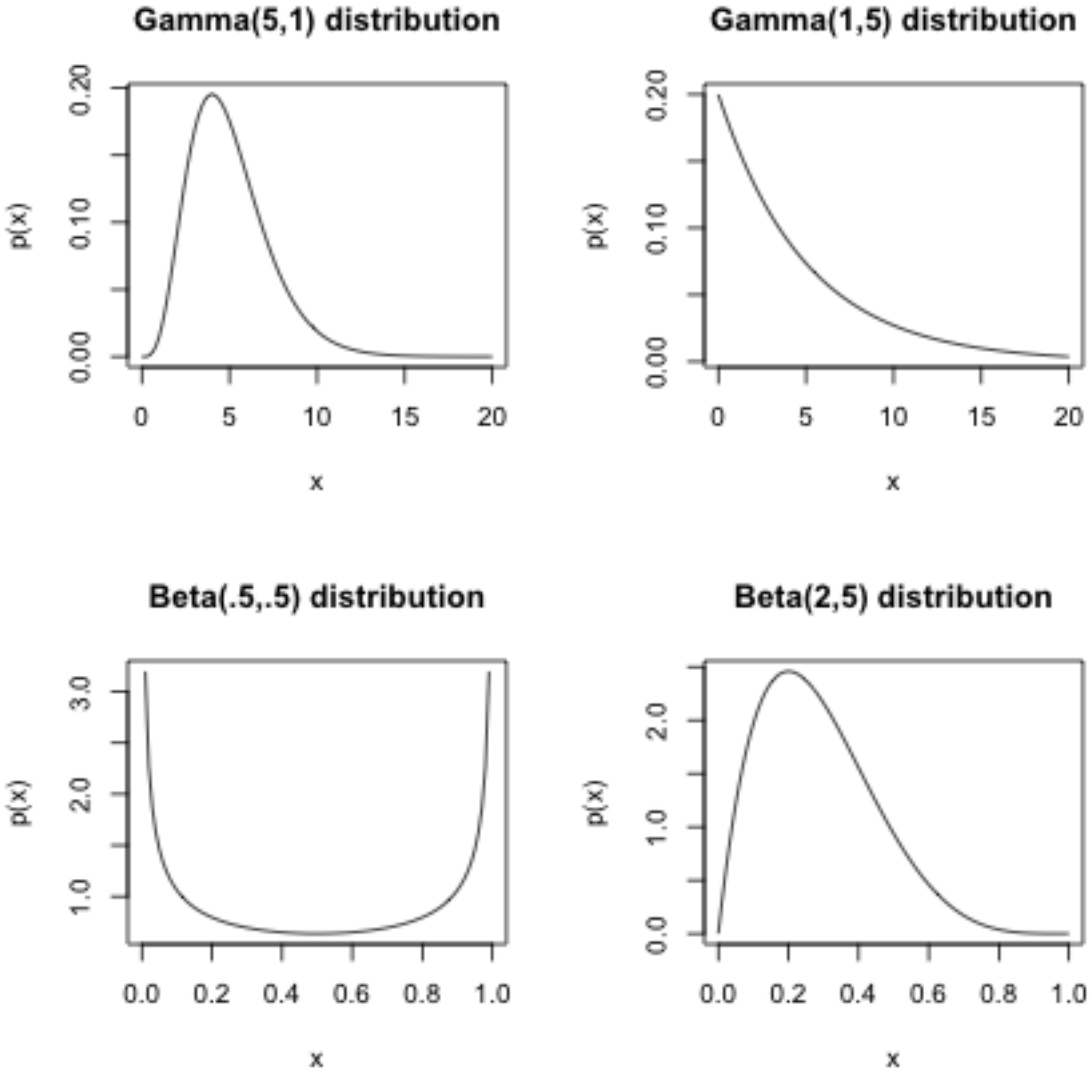
```
m <- mean(flightSF_nc$DepDelay)
s <- sqrt(var(flightSF_nc$DepDelay)/sampleSize)
p <- function(x) {
    dnorm(x, mean = m, sd = s)
}
par(mfrow = c(2, 2))
hist(sampleMean, xlab = "Mean Departure Delay", main = expression(paste(bar(X),
    ", default breaks")), freq = FALSE)
curve(p, add = TRUE, col = "red", lwd = 3)
hist(sampleMean, xlab = "Mean Departure Delay", main = expression(paste(bar(X),
    ", 10 breaks")), breaks = 10, freq = FALSE)
curve(p, add = TRUE, col = "red", lwd = 3)
hist(sampleMean, xlab = "Mean Departure Delay", main = expression(paste(bar(X),
    ", 1000 breaks")), freq = FALSE, breaks = 1000)
curve(p, add = TRUE, col = "red", lwd = 3)
hist(sampleMean, xlab = "Mean Departure Delay", main = expression(paste(bar(X),
    ", 10000 breaks")), freq = FALSE, breaks = 10000)
curve(p, add = TRUE, col = "red", lwd = 3)
```



4.2.3 Other distributions

Here are some examples of some pdfs from some two common continuous distributions other than the normal:

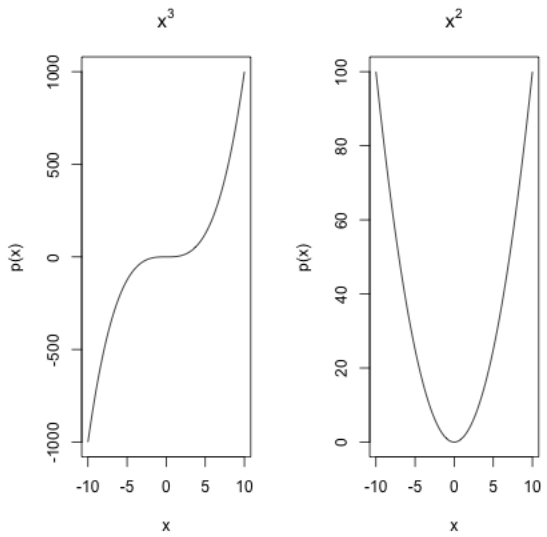
```
par(mfrow = c(2, 2))
f <- function(x) {
    dgamma(x, shape = 5, scale = 1)
}
curve(f, from = 0, to = 20, ylab = "p(x)", main = "Gamma(5,1) distribution")
f <- function(x) {
    dgamma(x, shape = 1, scale = 5)
}
curve(f, from = 0, to = 20, ylab = "p(x)", main = "Gamma(1,5) distribution")
f <- function(x) {
    dbeta(x, 0.5, 0.5)
}
curve(f, from = 0, to = 1, ylab = "p(x)", main = "Beta(.5,.5) distribution")
f <- function(x) {
    dbeta(x, 2, 5)
}
curve(f, from = 0, to = 1, ylab = "p(x)", main = "Beta(2,5) distribution")
```



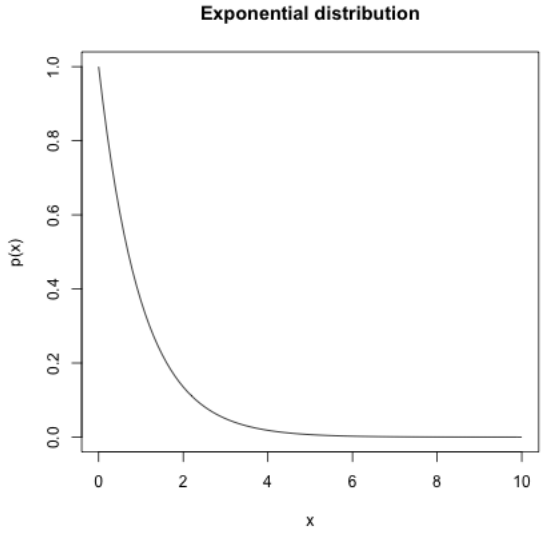
Notice a few things illustrated by these examples:

- that 'a' distribution actually can be multiple distributions that differ by changing the parameters (e.g. Normal has a mean and a standard deviation that defines it)
- Unlike the normal, many distributions have very different shapes for different parameters
- Continuous distributions can be concentrated to an interval or region (i.e. not take on all values of the real line). They are still considered continuous distributions because they range of points with positive probability is still a continuous range.

The following cannot be pdfs, why?



But be careful. Just because a function $p(x)$ goes to infinity, doesn't mean that it can't be a probability density!



4.3 Density Curve Estimation

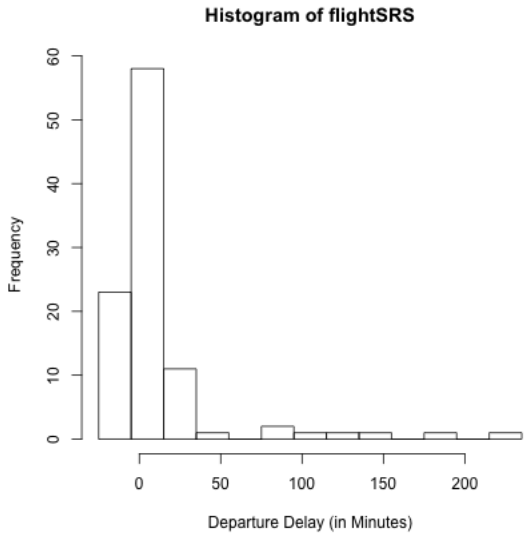
We've seen that histograms can approximate density curves (by making the area in the histogram sum to 1). If we have data from a continuous distribution, we are estimating a pdf, so we would want an estimate that is written as a function, say $\hat{p}(x)$.

Histogram as estimate of pdf For a continuous distribution we can only calculate probabilities in small interval around x. If the pdf is pretty smooth, then in a small window around x, $p(x)$ is going to be roughly the same value, so that if width of the interval is small, the probability in a small interval around x is roughly proportional to $p(x)$.[8] So if we want to estimate $p(x)$, we could estimate the probability of a small interval if width w around x, how?

Then the true probability of that interval is roughly $wp(x)$ (because we are assuming pdf is smooth, so not changing much). So we could say

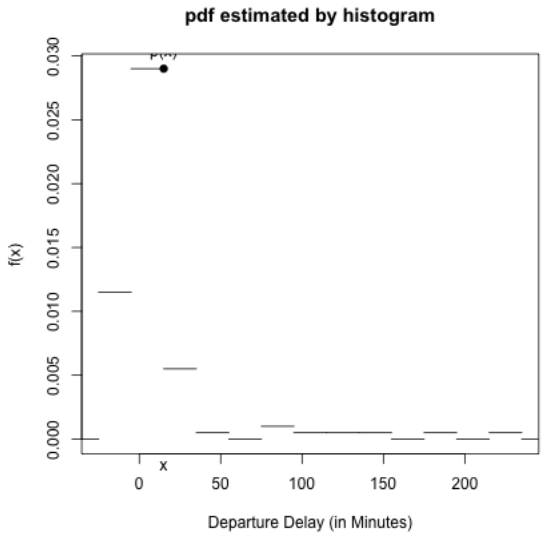
$$\hat{p}(x) = \frac{\hat{P}(\text{data in interval})}{w}$$

With this idea, we can view our histogram as a estimate of the pdf. For example, suppose we consider a histogram of our SRS of flights from January,



[8] Don't forget you have to take into account the width of the interval, so its not actually $p(x)$, but $p(x)$ times a very small amount!

Then the frequency counts in the frequency version of a histogram can be convert to density scale by dividing by the width of the interval of the bins (this is what is meant by the density values in a histogram). Then by our argument above, this is an estimate of $p(x)$, specifically an estimate $\hat{p}(x)$ that is what is called a step function:



Basically, interpreting a histogram as an estimate of the pdf means, in the interval of the bin, we assume $p(x)$ is roughly the same and estimated by

$$\hat{p}_{hist}(x \in \text{bin}) = \frac{\hat{P}(\text{data in bin})}{w}$$

So if x is in the bin with interval $[5, 7)$, then if w is the width of the bin, how do you calculate $\hat{p}_{hist}(4)$?

4.3.1 Kernel density estimation

The histogram estimate is reasonable estimate if x is right in the middle of the bin, but if x is on the boundary of the bin, what happens?

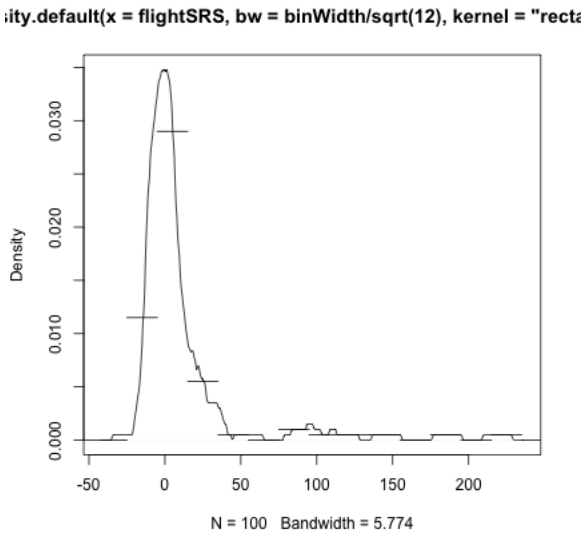
It makes not only the *size* of the bins, but also the specific *centers* of the bins important. And clearly this doesn't make for a continuous function!

Moving Windows We could imagine making different intervals for each value of x, so that for the interval used to estimate $p(x)$ is centered at x.

For example, say we pick a bin width of 2, and want to estimate the density around 5. Then for $x = 5$, we could make a interval $[4, 6)$, and calculate

$$\frac{\#\text{ x } \in [4, 6)}{2 \times 100}$$

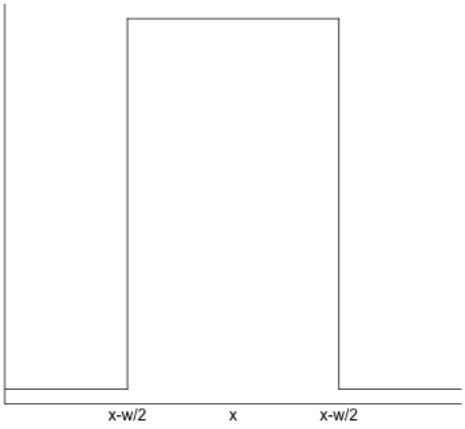
We can do this for $x = 6$, with an interval of $[5, 7)$ and so forth for each x. This would create a curve that looks like this.



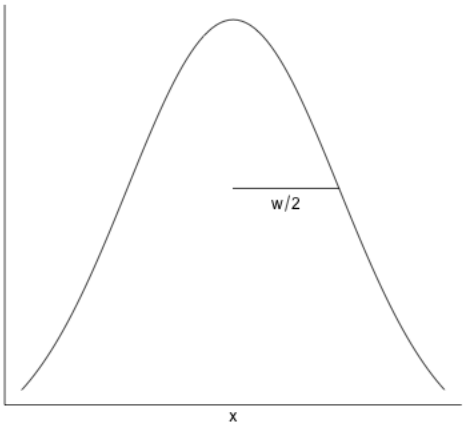
Moving Windows as a Weighted Kernel Function Our estimate of $p(x)$, more generally, is

$$\hat{p}(x) = \frac{\#x_i \in [x - \frac{w}{2}, x + \frac{w}{2})}{w \times n}$$

So to estimate the density around x, we are using the individual data observations if and only if they are close to x. Here is a visualization of how we determine whether a point x_i should contribute to estimating $p(x)$



Once we think about it like that, we can think about not having such a sharp distinction for the interval around x. How much you contribute to the estimate of $p(x)$ could be based on your distance from x, but in a smooth way. For example, consider this more 'gentle' visualization of the contribution of x_i:



We call both of these functions a **kernel function** and are ways to choose how to let nearby points contribute to the estimate of x.

The second one is normal (or gaussian) kernel and is very common for density estimation. It is a normal curve centered at x[9]; as you move away from x you start

[9]You have to properly scale the height of the kernel function curve so that you get area under the final estimate $\hat{p}(x)$ curve equal to 1

to decrease in your contribution to the estimate of $p(x)$ but more gradually than the rectangle kernel we started with.

Writing this as an equation We can re-write counting as a sum of a series of 0-1 decisions about each point x_i, often written as an indicator function, $I(\cdot)$, meaning the value of $I(\cdot) = 1$ if the expression $(\cdot)$ inside is true, and zero otherwise.

$$\#x_i \in [x - \frac{w}{2}, x + \frac{w}{2}) = \sum_{i=1}^{n} I(x - \frac{w}{2} \leq x_i \leq x + \frac{w}{2})$$

Then our estimate of $p(x)$ can be written as

$$\hat{p}(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{w} I(|x_i - x| \leq \frac{w}{2})$$

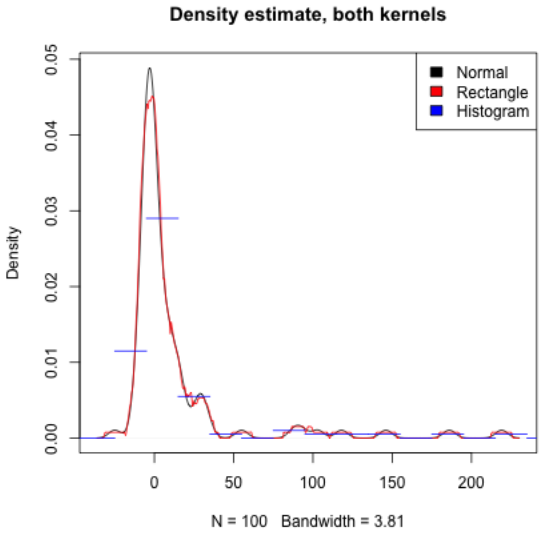
(note I've changed $x - \frac{w}{2} \leq x_i \leq x + \frac{w}{2}$ into a more succinct $|x_i - x| \leq \frac{w}{2}$)

So to estimate the density around x, we are using the individual data observations if and only if they are close to x, and $I_x(\cdot)$ is the function that controls that.

So to use the gaussian kernel above instead, we want to substitute $R_x(x_i) = \frac{1}{w} I(|x_i - x| \leq \frac{w}{2})$ with a (properly) scaled function $f_x(x_i)$ that is the normal pdf.

Example of Flight data Here is the estimate of the density based on the rectangular kernel and the normal kernel, along with our estimate from the histogram:

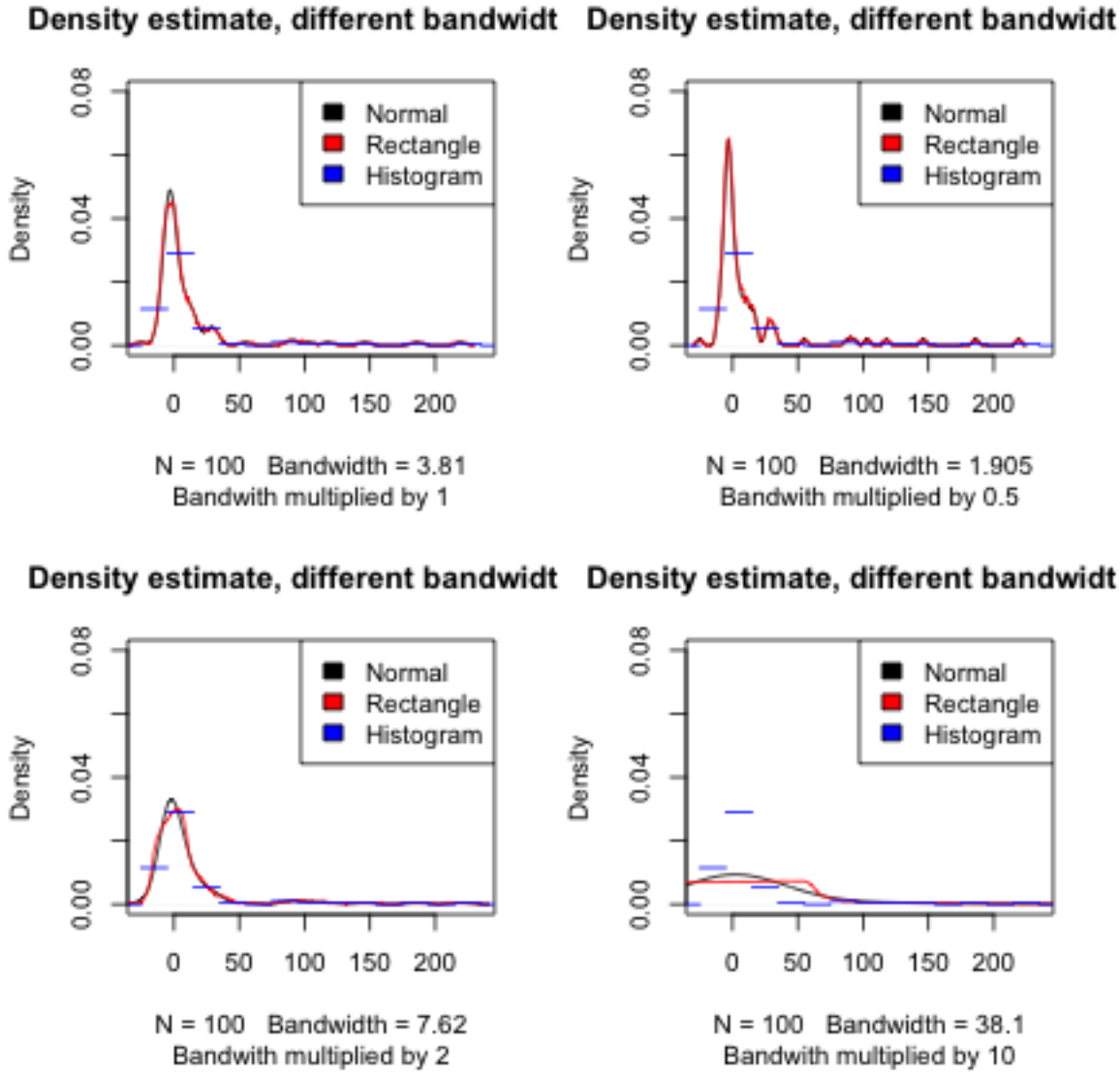
```
plot(density(flightSRS, kernel = "gaussian"), main = "Density estimate, both kernels"
lines(density(flightSRS, kernel = "rectangular"), col = "red")
lines(p, do.points = FALSE, xlab = "Departure Delay (in Minutes)",
    verticals = FALSE, ylim = ylim, xlim = xlim, col = "blue")
legend("topright", c("Normal", "Rectangle", "Histogram"),
    fill = c("black", "red", "blue"))
```



What do you notice when comparing the estimates of the density from these two kernels?

Bandwidth Notice that I still have a problem of picking a width for the rectangular kernel, or the spread/standard deviation for the gaussian kernel. This w is called generically a **bandwidth** parameter. In the above plot I let the `density` function pick it in an automatic way.

Here are different choices of the bandwidth:

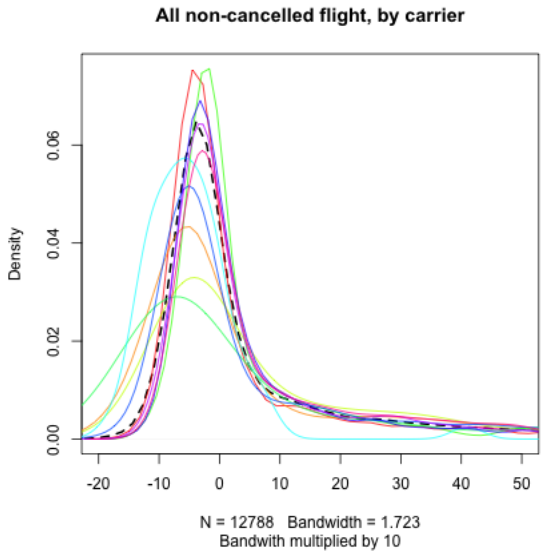


4.3.2 Comparing multiple groups with density curves

In addition to being a more satisfying estimation of a pdf, density curves are much easier to compare between groups than histograms because you can easily overlay them.

```
perGroupDensity <- tapply(X = flightSF_nc$DepDelay,
    INDEX = flightSF_nc$Carrier, FUN = density)
ylim <- range(sapply(perGroupDensity, function(x) {
    range(x$y)
}))
cols <- rainbow(length(perGroupDensity))
```

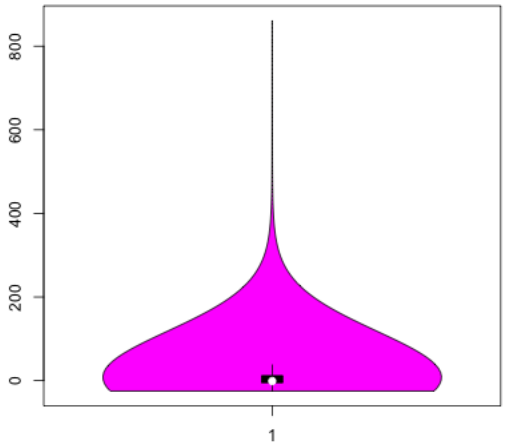
```
par(mfrow = c(1, 1))
plot(density(flightSF_nc$DepDelay), main = "All non-cancelled flight, by carrier",
    sub = paste("Bandwith multiplied by", adjust),
    lwd = 2, lty = 2, xlim = c(-20, 50), ylim = ylim)
nullOut <- mapply(perGroupDensity, cols, FUN = function(x,
    col) {
    lines(x, col = col)
})
```



4.3.3 Violin Plots

We can combine the idea of density plots and boxplots to get something called a 'violin plot'.

```
library(vioplot)
vioplot(flightSF_nc$DepDelay)
```



This is basically just turning the density estimate on its side and putting it next to the boxplot so that you can get finer-grain information about the distribution. Like boxplots, this allows you to compare many groups (but unlike the standard `boxplot` command, the `vioplot` function is a bit akward for plotting multiple groups, so I've made my own little function 'vioplot2' available online which I will import here)

```
source("http://www.stat.berkeley.edu/~epurdom/RcodeForClasses/myvioplot.R")
vioplot2(flightSF_nc$DepDelay, flightSF_nc$Carrier,
    col = palette())
```