Statistics in R

An introduction

Kevin Rue-Albrecht
University of Oxford
2020-04-03 (updated: 2020-04-23)

Overview

- Statistical distributions available in R
- Working with statistical distributions
- Perform statistical tests and inspect results
- Permutation, resampling, and bootstraping

R is built for statistics

- R includes a number of common statistical distributions:
  - The Normal Distribution
  - The Binomial Distribution
  - The Poisson Distribution
  - ...
- R implements a range of statistical tests:
  - Student's t-Test
  - Pearson's Chi-squared Test for Count Data
  - Wilcoxon Rank Sum and Signed Rank Tests
  - ...

Functions for Probability Distributions

Distribution	Probability	Quantile	Density	Random
Beta	`pbeta`	`qbeta`	`dbeta`	`rbeta`
Binomial	`pbinom`	`qbinom`	`dbinom`	`rbinom`
Cauchy	`pcauchy`	`qcauchy`	`dcauchy`	`rcauchy`
Chi-Square	`pchisq`	`qchisq`	`dchisq`	`rchisq`
Exponential	`pexp`	`qexp`	`dexp`	`rexp`
F	`pf`	`qf`	`df`	`rf`
Gamma	`pgamma`	`qgamma`	`dgamma`	`rgamma`
Geometric	`pgeom`	`qgeom`	`dgeom`	`rgeom`
Hypergeometric	`phyper`	`qhyper`	`dhyper`	`rhyper`
Logistic	`plogis`	`qlogis`	`dlogis`	`rlogis`
Log Normal	`plnorm`	`qlnorm`	`dlnorm`	`rlnorm`
Negative Binomial	`pnbinom`	`qnbinom`	`dnbinom`	`rnbinom`
Normal	`pnorm`	`qnorm`	`dnorm`	`rnorm`
Poisson	`ppois`	`qpois`	`dpois`	`rpois`
Student t	`pt`	`qt`	`dt`	`rt`
Studentized Range	`ptukey`	`qtukey`	`dtukey`	`rtukey`
Uniform	`punif`	`qunif`	`dunif`	`runif`
Weibull	`pweibull`	`qweibull`	`dweibull`	`rweibull`
Wilcoxon Rank Sum Statistic	`pwilcox`	`qwilcox`	`dwilcox`	`rwilcox`

- Each distribution has a root name, e.g. `norm`
- Every distribution has four functions.
- The root name is prefixed by one of the letters:
  - `p` for "probability", the cumulative distribution function (c. d. f.)
  - `q` for "quantile", the inverse c. d. f.
  - `d` for "density", the density function (p. f. or p. d. f.)
  - `r` for "random", a random variable having the specified distribution

The normal distribution

Notation

$$\mathcal{N}(\mu, \sigma^2)$$

Parameters

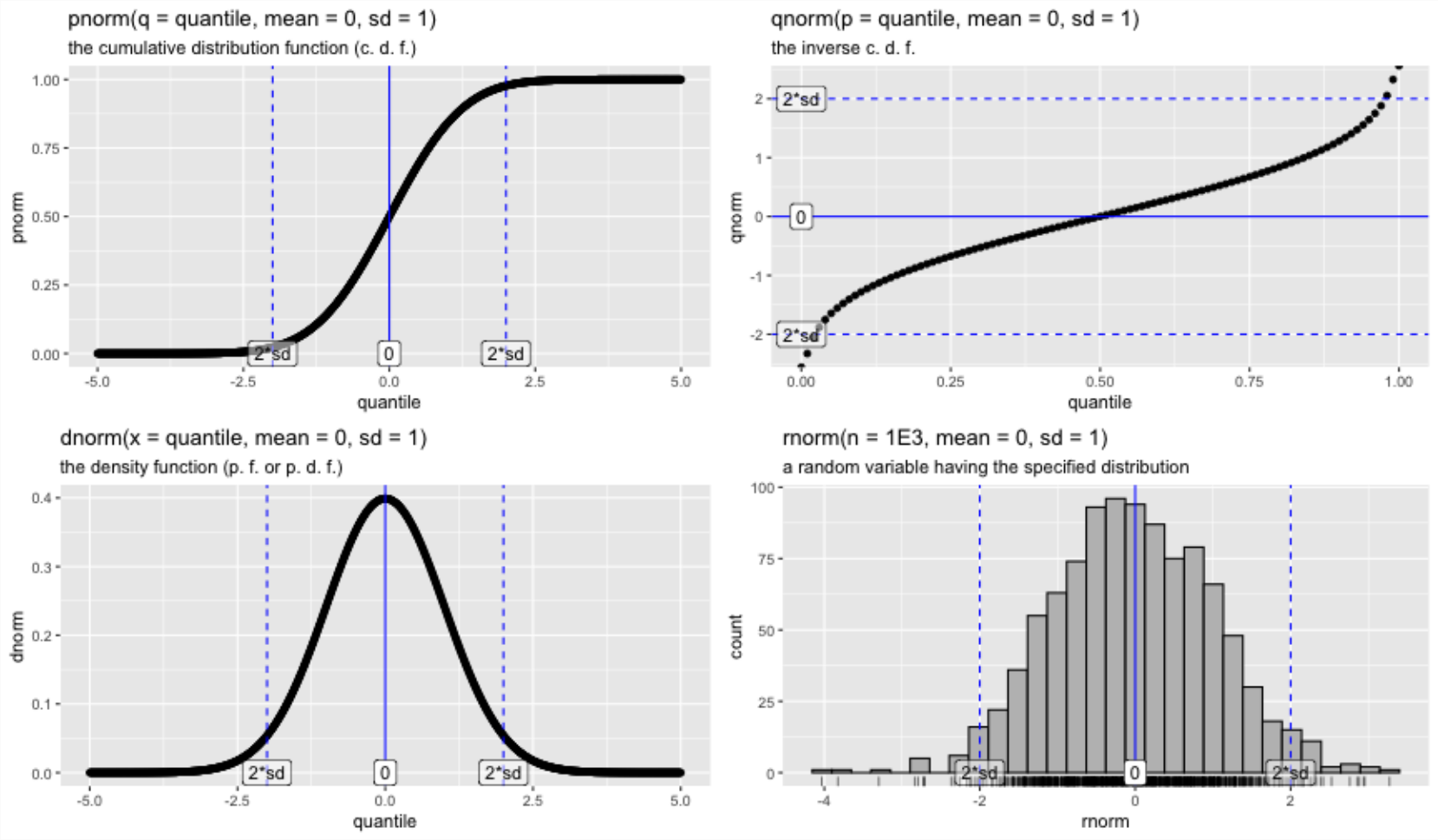
- $\mu \in \mathbb{R}$ = mean (location)
- $\sigma^2 > 0$ = variance (squared scale)

Properties

- Median: μ
- Mode: μ
- Probability density function (PDF): $\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$
- Variance: σ^2

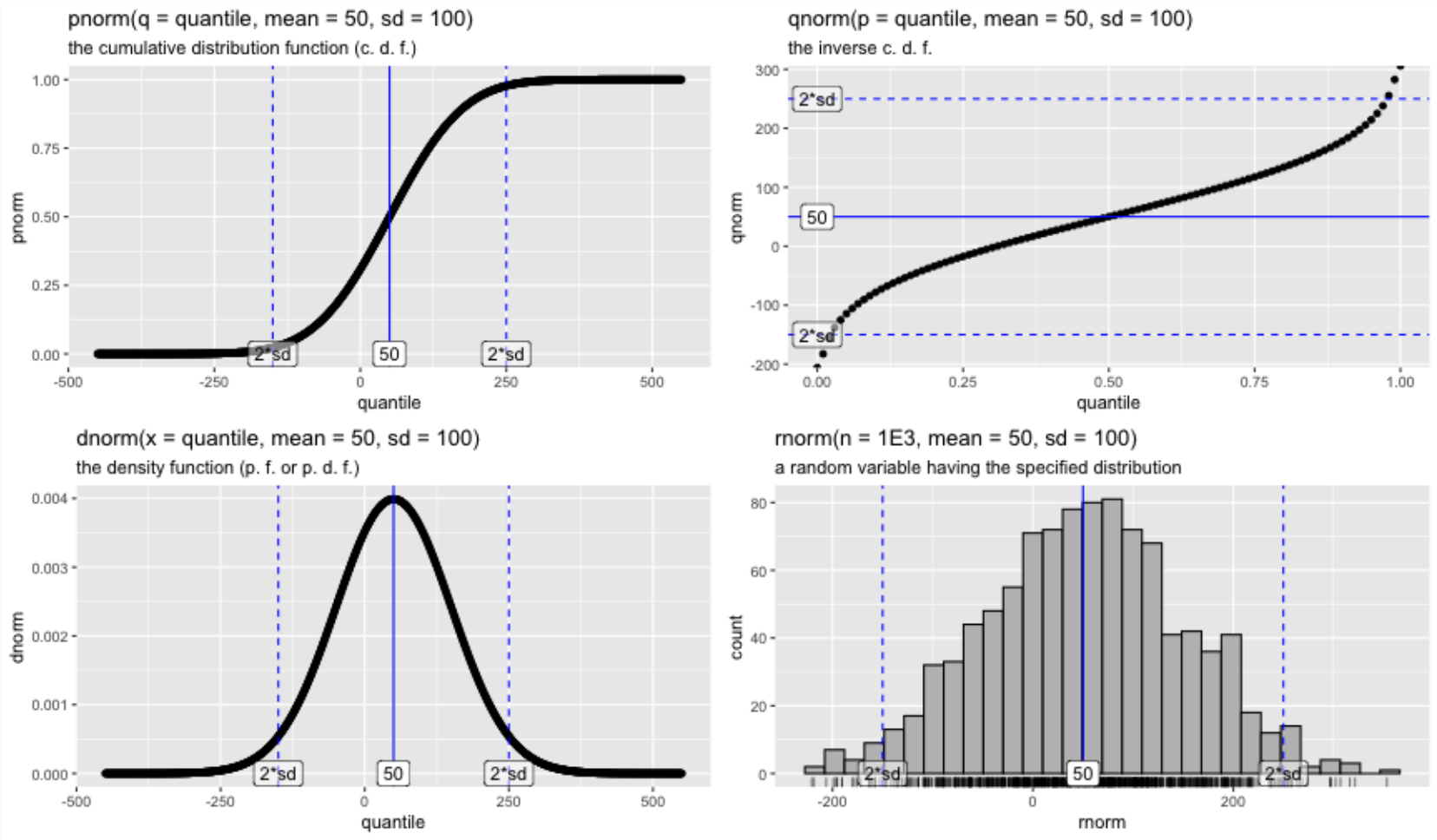
The standard normal distribution (mean: 0, sd: 1)

Standard normal distribution with mean 0 and standard deviation 1.



A parameterised normal distribution (mean: 50, sd: 100)

Normal distribution parameterised with mean 50 and standard deviation 100.



The normal distribution: `rnorm`

`rnorm` is the R function that simulates random variates having a specified normal distribution.

For instance, `rnorm` can be used to generate a vector of 1,000 values, normally distributed with a mean of 0 and a standard deviation of 1.

```
norm_vector <- rnorm(n = 1000, mean = 0, sd = 1)
```

`summary()` is useful to inspect the properties of values.

```
summary(norm_vector)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -2.98317 -0.64935  0.01436  0.01626  0.62990  3.61254
```

`quantile()` returns quantiles of the sample corresponding to given probabilities.

```
quantile(x = norm_vector, probs = seq(from = 0, to = 1, length.out = 5))
```

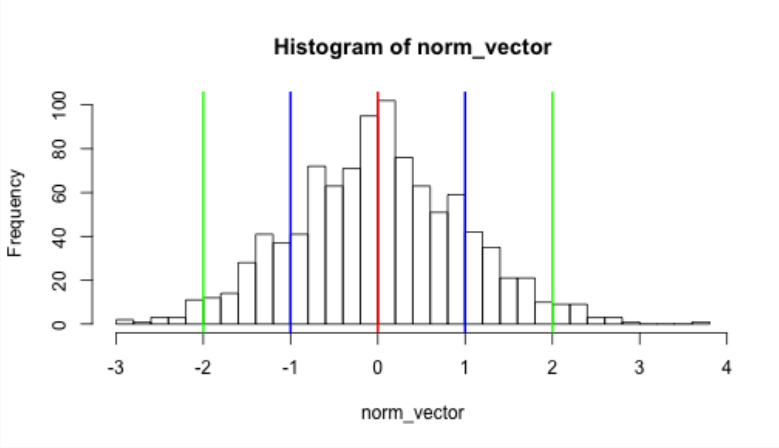
```
##          0%         25%         50%         75%        100%
## -2.98317028 -0.64935421  0.01436526  0.62989671  3.61254021
```

The normal distribution: visualise with base R

Visualisation

`hist` is a useful base R function to inspect the distribution of values.

```
hist(norm_vector, breaks = 30)
abline(v = 0, col="red", lwd=2)
abline(v = c(-1, 1), col="blue", lwd=2)
abline(v = c(-2, 2), col="green", lwd=2)
```



Properties

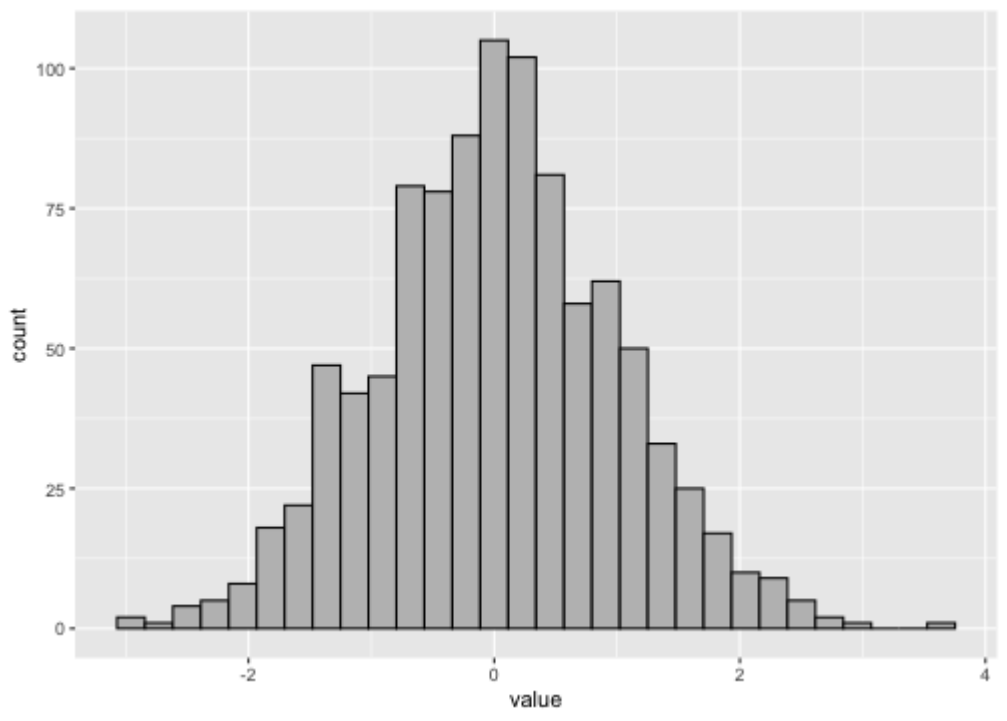
- The mean of those randomly generated values is close to 0
- The standard deviation of those randomly generated values is close to 1
- ~ 64% of the data points are within 1 standard deviation of the mean (blue lines)
- ~ 95% of the data points are within 2 standard deviations of the mean (green lines)

Exercise: Check the above properties.

The normal distribution: visualised with *ggplot2*

`ggplot` requires the data formatted as a table first, but generates better-looking graphics.

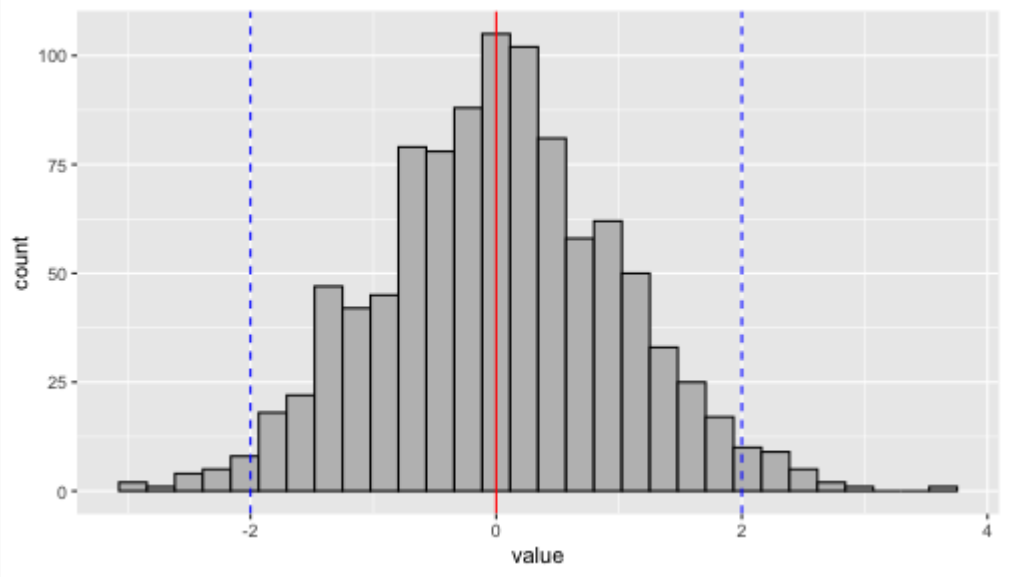
```
ggplot() + geom_histogram(
    aes(x = value), data = tibble(value = norm_vector),
    bins = 30, color = "black", fill = "grey")
```



The normal distribution: visualised with *ggplot2*

`ggplot` requires the data formatted as a table first, but generates better-looking graphics.

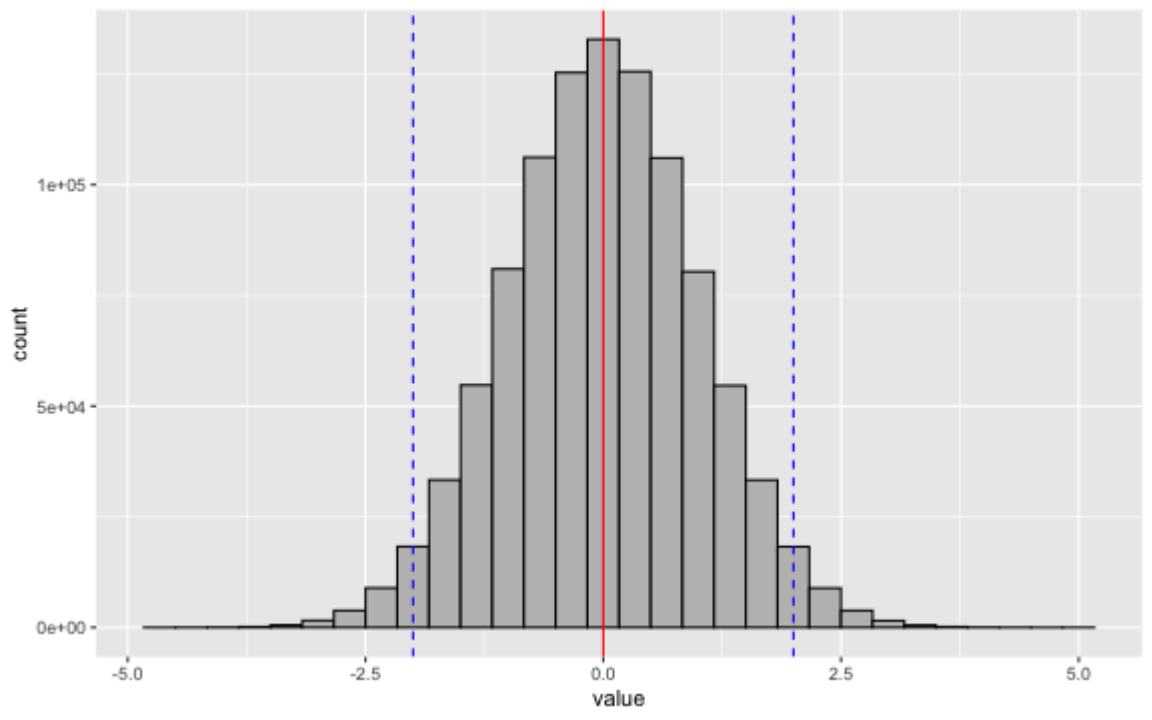
```
ggplot() + geom_histogram(
  aes(x = value), data = tibble(value = norm_vector),
  bins = 30, color = "black", fill = "grey") +
  geom_vline(xintercept = 0, color = "red", linetype = "solid") +
  geom_vline(xintercept = c(-2, 2), color = "blue", linetype = "dashed")
```



The normal distribution: more data points

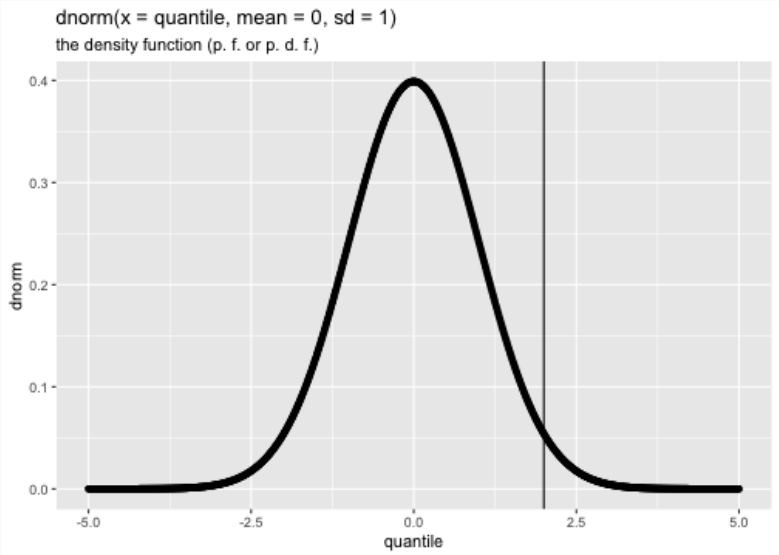
More data points build better models.

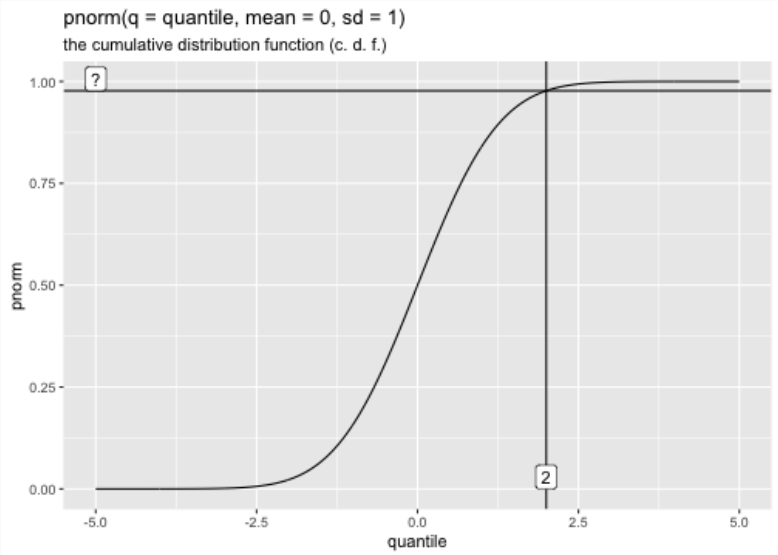
```
norm_vector <- rnorm(n = 1000000, mean = 0, sd = 1)
```



The normal distribution: "what is the probability" ?

Exercise: For data originating from the standard normal distribution, what is the probability of observing a value greater than 2?





Hint: Use `pnorm`.

Hypothesis testing: the `iris` dataset

Every time we want to test an assumption regarding a population parameter in a dataset.

For instance, we can use the `iris` dataset.

```
data("iris")
tibble(iris)
```

```
## # A tibble: 150 x 5
##    Sepal.Length Sepal.Width Petal.Length Petal.Width Species
##           <dbl>       <dbl>        <dbl>       <dbl> <fct>
##  1          5.1         3.5          1.4         0.2 setosa
##  2          4.9         3            1.4         0.2 setosa
##  3          4.7         3.2          1.3         0.2 setosa
##  4          4.6         3.1          1.5         0.2 setosa
##  5          5           3.6          1.4         0.2 setosa
##  6          5.4         3.9          1.7         0.4 setosa
##  7          4.6         3.4          1.4         0.3 setosa
##  8          5           3.4          1.5         0.2 setosa
##  9          4.4         2.9          1.4         0.2 setosa
## 10          4.9         3.1          1.5         0.1 setosa
## # … with 140 more rows
```

Hypothesis testing: dataset summary

For tables, `summary()` reports summary statistics for each column.

```
summary(iris)
```

```
##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width   
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100  
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300  
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300  
##  Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199  
##  3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800  
##  Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500  
##        Species  
##  setosa    :50  
##  versicolor:50  
##  virginica :50  
##                 
##                 
##
```

There are 3 species in the dataset:

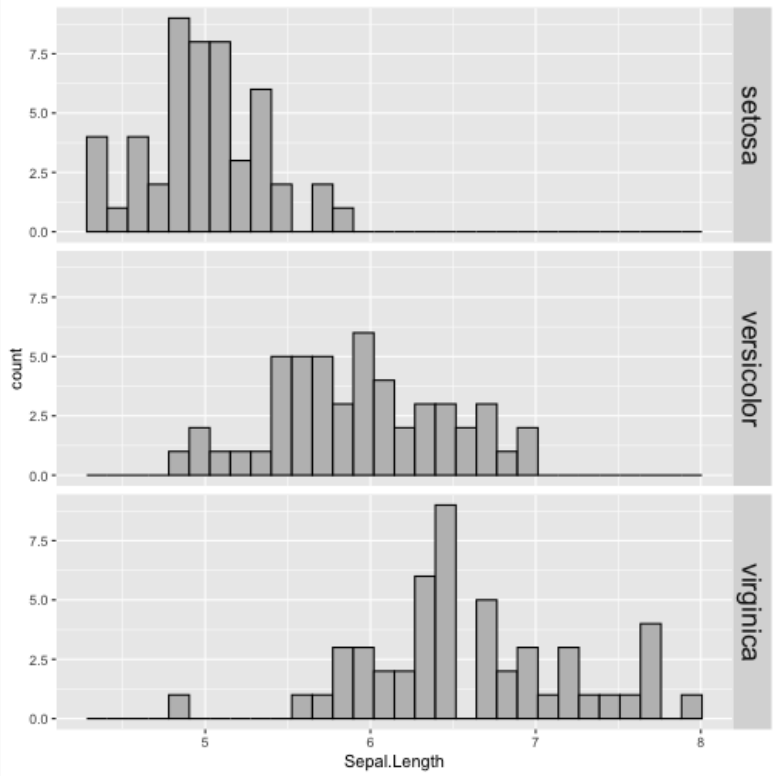
- setosa
- versicolor
- virginica

Each of them has 50 observations.

Hypothesis testing: visualise the data

What does the distribution of sepal length look like for the various species?

```
ggplot(iris) +
  geom_histogram(
    aes(x = Sepal.Length),
    bins = 30,
    color = "black",
    fill = "grey") +
  facet_grid(Species ~ .) +
  theme(
    strip.text.y=element_text(size=rel(2))
  )
```



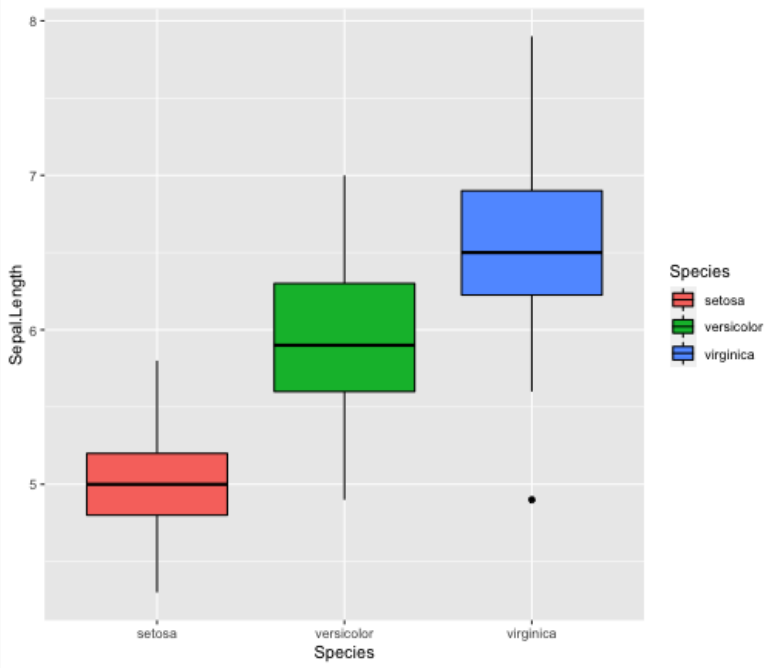
Those distribution *look* different. However, are those differences *statistically significant*?

Hypothesis testing: visualise the data

What does the distribution of sepal length look like for the various species?

```
ggplot(iris) +
  geom_boxplot(
    aes(
      x = Species, y = Sepal.Length,
      fill = Species),
    color = "black")
```

What does the distribution of sepal length look like for the various species?



Those distribution *look* different. However, are those differences *statistically significant*?

Hypothesis testing: Apply Student's t-Test

Is the difference of sepal length between species `setosa` and `versicolor` *significant*?

```
df_test <- iris %>%
  select(Sepal.Length, Species) %>%
  filter(Species == "setosa" | Species == "versicolor")
t.test(Sepal.Length ~ Species, data = df_test)
```

```
##
##     Welch Two Sample t-test
##
## data:  Sepal.Length by Species
## t = -10.521, df = 86.538, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.1057074 -0.7542926
## sample estimates:
##     mean in group setosa mean in group versicolor
##                    5.006                    5.936
```

Answer: In this case, yes!

Credits

- `xaringan` package by Yihui Xie, used to make these slides.
- `rladies` slide theme by Alison Hill.
- Charlotte Soneson for sharing pointers to online materials.

Further reading

- UCLouvain Bioinformatics Summer School 2019
  - Introduction to Statistics and Machine Learning by Oliver M. Crook
  - Practical: stats/ML
- CSAMA by the European Molecular Biology Laboratory (EMBL).
- Statistic with R and dplyr and ggplot by Greg Martin