

Week 2 Module 6: Pie Charts

CSCI E-5a: Programming in R

Let's clear the global computing environment:

```
rm( list = ls() )
```

Module Overview and Learning Objectives

In this module, we'll learn what *pie charts* are, and how to create them.

- In Section 1, we'll define the concept of a pie chart, and discuss some of the advantages and disadvantages of this visualization method.
- In Section 2, we'll construct a basic pie chart.
- In Section 3, we'll modify the basic pie chart.
- In Section 4, we'll explore some strategies for selecting colors for a pie chart.

When you've completed this module, you'll be able to:

- Create a basic pie chart given a set of numeric values.
- Modify the pie chart by explicitly specifying labels, titles, and colors.

We'll meet one new built-in R function.

- `pie()`

All right! Let's get started by defining the concept of a pie chart.

Section 1: What's a "Pie Chart"?

Main Idea: *We can define the structure of pie charts.*

In this section, we'll define the concept of a pie chart, and discuss some of the advantages and disadvantages of this visualization method.

Now let's learn our first data visualization method: the pie chart.

The pie chart is useful when we have a collection of categories, each with an associated numeric value, and we want to visualize the relative proportions of this numeric value across the categories.

For instance, a company could display the proportion of its operating budget in terms of sales, marketing, product development, manufacturing, and technical support.

The pie chart presents these relative proportions by displaying a circle (i.e. the “pie”) and then showing each category as a colored wedge (i.e. the “pie slices”).

Categories with larger associated values would have larger slices, and so it’s easy for a viewer to visualize the relative proportions of the different categories.

A pie chart only makes sense when the categories form a *partition* of the data – that is, when we have a collection of objects that we can split up into different categories.

For instance, it’s fine to make a pie chart for the number of hits for a baseball batter, because the set of hits can be partitioned into singles, doubles, triples, and home runs.

However, it wouldn’t make sense to create a pie chart for batting average, on-base percentage, and slugging average, because these values don’t constitute a partition of anything.

Many people are critical of pie charts, but I’m not convinced by the arguments that I’ve seen.

The R documentation of the `pie` function says this:

“Pie charts are a very bad way of displaying information. The eye is good at judging linear measures and bad at judging relative areas. A bar chart or dot chart is a preferable way of displaying this type of data.”

So even the authors of the official documentation don’t like pie charts!

However, I think this misses the point.

While in general our eyes are not good at judging relative areas, because of the circular layout of the pie chart the relative areas are equivalent to the relative angle sizes, and that is something that can be judged by eye.

In the real world, pie charts are used frequently to communicate information, so it must be a useful method, regardless of what the R documentation says.

Personally, I think that pie charts are pleasant to look at, and you should keep them in your repertoire of data visualizations if for no other reason than visual variety.

But of course you should use them only when they are appropriate.

Ultimately, deciding to use or not to use a pie chart is a matter of subjective taste and personal preference.

So, that’s what a pie chart is.

Now let’s work on an example.

Section 2: Making a Pie Chart

Main Idea: *We can construct a pie chart.*

In this section, we’ll construct a basic pie chart.

Suppose we are presenting an analysis of an annual budget with these components:

Component	Costs
Sales	9.8
Marketing	4.2

Component	Costs
Research	3.1
Manufacturing	2.8
Technical Support	1.7
Total	21.6

How can we visualize the relative proportions of these costs?

The pie slice for Sales should be the largest, the pie slice for Marketing should be the next largest and a little less than 1/2 the area of the Sales slice, and so on.

To make the pie chart, we have to construct a numeric vector consisting of the component costs.

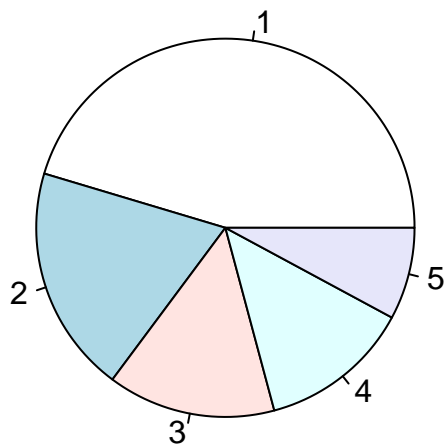
We can do that with the `c()` function:

```
c( 9.8, 4.2, 3.1, 2.8, 1.7)
```

```
## [1] 9.8 4.2 3.1 2.8 1.7
```

Once we have the vector, we make a simple pie chart by using the `pie()` function and passing it the data:

```
pie(
  x = c( 9.8, 4.2, 3.1, 2.8, 1.7 )
)
```



Notice that we don't have to calculate the relative proportions of the numeric values – the `pie()` function takes care of that for us.

So that's how to make a basic pie chart.

Now let's see how to modify this pie chart to make it look more finished.

Exercise 5.1: Babe Ruth At-Bats

Recall that an at-bat is either an out or a hit, and there are four different kinds of hits.

Thus, there are five possible outcomes for an at-bat:

- An out
- A single
- A double
- A triple
- A home run

Let's make a simple pie chart to display the relative proportions of these outcomes for Babe Ruth.

Recall that Babe Ruth had these batting statistics:

Statistics	Value
Plate appearances	10,626
At-bats	8,399
Hits	2,873
Doubles	506
Triples	136
Home runs	714
Bases on balls	2,062
Hit by a pitch	43
Sacrifice flies	0

Remember these two derived statistics:

- To calculate the number of hits, subtract the number of hits from the number of at-bats
- To calculate the number of singles, subtract the number of doubles, triples, and home runs from the number of hits.

Here are some variables for you:

```
babe.ruth.plate.appearances <- 10626  
  
babe.ruth.at.bats <- 8399  
  
babe.ruth.hits <- 2873  
babe.ruth.doubles <- 506  
babe.ruth.triples <- 136  
babe.ruth.home.runs <- 714
```

Solution

Section 3: Modifying the Pie Chart

Main Idea: *We can modify the basic pie chart.*

In this section, we'll modify the basic pie chart.

Now our simple pie chart isn't a bad start, but it can definitely be improved.

For one thing, although the pie slices are in the right proportions, the graph doesn't indicate which component each slice represents – which one is Sales, and which one is Technical Support, etc.?

To do this, we have to specify an optional argument named `labels` by using a character vector consisting of the character strings for the respective component:

```
# Example 3: Creating a vector of labels
```

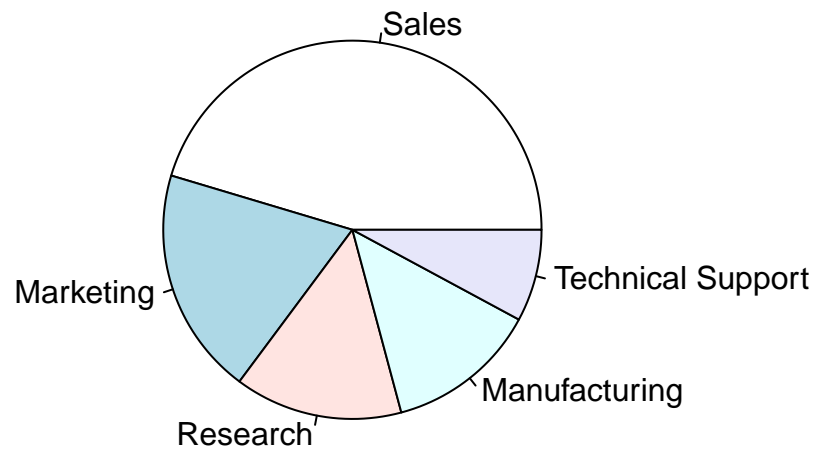
```
c( "Sales", "Marketing", "Research",  
   "Manufacturing", "Technical Support" )
```

```
## [1] "Sales"          "Marketing"        "Research"  
## [4] "Manufacturing"   "Technical Support"
```

Now when we call the `pie()` function we supply both the vector of costs and the vector of component names:

```
# Example 4: A pie chart with labels
```

```
pie(  
  x = c( 9.8, 4.2, 3.1, 2.8, 1.7),  
  labels =  
    c( "Sales", "Marketing", "Research",  
       "Manufacturing", "Technical Support" )  
)
```



We can also include a title for our pie chart using the `main =` option:

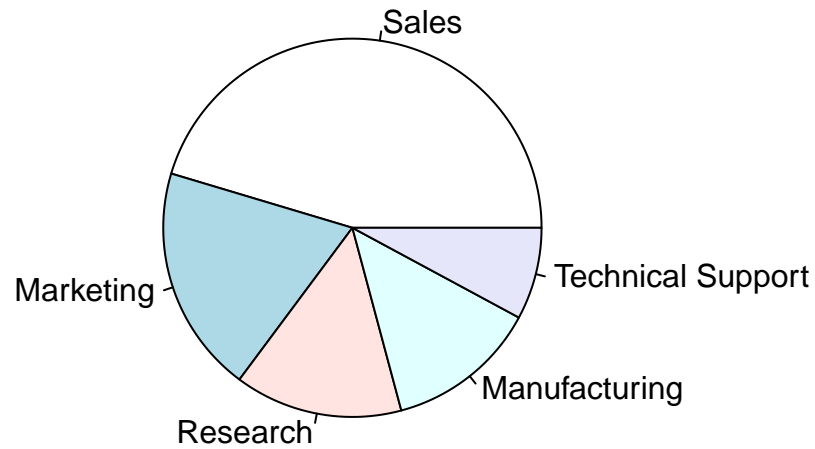
```
# Example 5: Pie chart with labels and main title

pie(
  c( 9.8, 4.2, 3.1, 2.8, 1.7),

  labels =
    c( "Sales", "Marketing", "Research",
        "Manufacturing", "Technical Support" ),

  main = "Distribution of costs for 2019 budget"
)
```

Distribution of costs for 2019 budget



If you look closely, you'll notice that the pie chart displays the pie slices in a counter-clockwise manner. We can display the pie slices in a clockwise manner by using the `clockwise = TRUE` option:

```
# Example 6: Clockwise pie chart

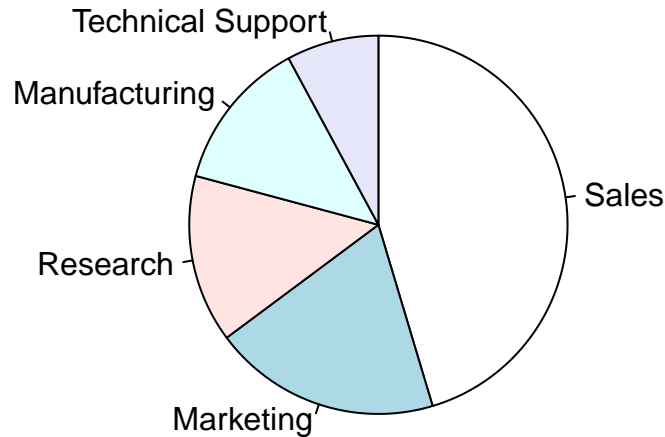
pie(
  c( 9.8, 4.2, 3.1, 2.8, 1.7),

  labels =
    c( "Sales", "Marketing", "Research",
      "Manufacturing", "Technical Support" ),

  main = "Distribution of costs for 2019 budget",

  clockwise = TRUE
)
```

Distribution of costs for 2019 budget



For our course, always use the clockwise option.

So that's how to modify the basic pie chart.

Now let's talk about some strategies for selecting colors for your pie chart.

Exercise 5.2: Modifying the Babe Ruth pie chart

Now add labels and a main title to the Babe Ruth pie chart, and set the rotation to clockwise.

Solution

Section 4: Pie Chart Colors

Main Idea: *We can select colors that work together for a pie chart.*

In this section, we'll explore some strategies for selecting colors for a pie chart.

We can also change the colors of the pie slices by using the `col` option.

We can specify the colors of the pie slices by constructing a character vector with the names of the desired colors.

This isn't always such a great option, because it takes some skill to find a good set of colors that work together.

For instance, I've chosen five colors for the pie chart, based on the fact that I think each color is attractive, at least when viewed alone:


```
# Example 7: Pie chart with garish colors

pie(
  x = c( 9.8, 4.2, 3.1, 2.8, 1.7),

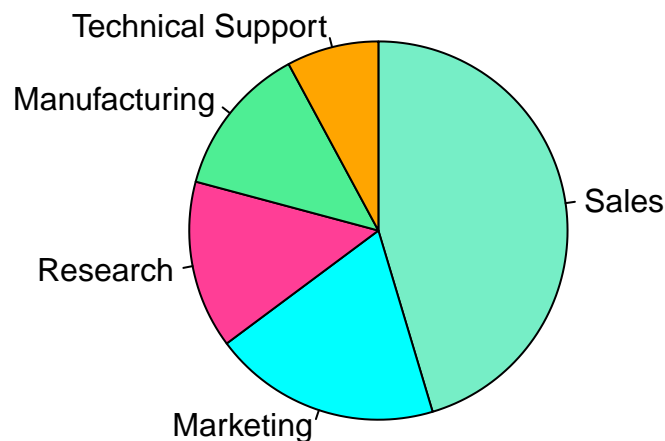
  labels =
    c( "Sales", "Marketing", "Research",
        "Manufacturing", "Technical Support" ),

  main = "Distribution of costs for 2019 budget",

  clockwise = TRUE,

  col = c("aquamarine2", "cyan", "violetred1",
          "seagreen2", "orange")
)
```

Distribution of costs for 2019 budget



Ugh.

This graph is ugly, because the colors don't work together as a set, regardless of how nice they might be in isolation.

Instead, when grouped together they are freakish and garish, the sort of thing that a clown would wear.

One strategy for finding colors that work well together is to use a color which has numbered variations.

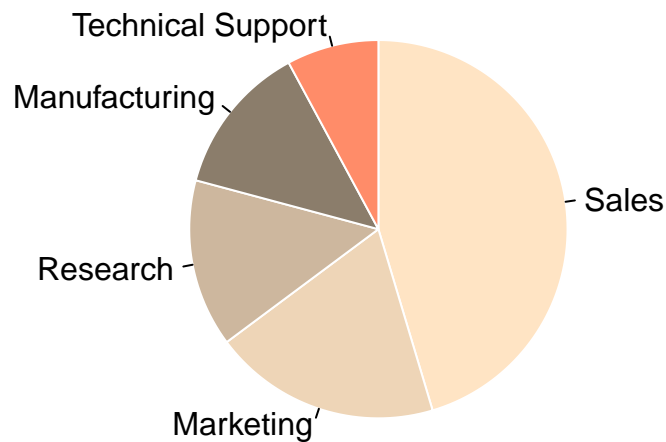
For instance, there are four varieties of the `bisque` color: `bisque1`, `bisque2`, `bisque3`, and `bisque4`.

I've use the brighter color `salmon1` to emphasize the Technical Support category:

Example 8: Pie chart with nicer colors and highlight

```
pie(  
  x = c( 9.8, 4.2, 3.1, 2.8, 1.7),  
  
  labels =  
    c( "Sales", "Marketing", "Research",  
      "Manufacturing", "Technical Support" ),  
  
  main = "Distribution of costs for 2019 budget",  
  
  col =  
    c( "bisque1", "bisque2", "bisque3",  
      "bisque4", "salmon1"),  
  
  clockwise = TRUE,  
  
  border = "white"  
)
```

Distribution of costs for 2019 budget



By the way, notice that I set the graphical option **border** to white to eliminate the lines.

Another strategy for finding a set of colors that work well together is to use the various shades of gray.

These will look good as a set, and you can use a true color to create emphasis.

For instance:

```
# Example 9: Pie chart with grayscale

pie(
  c( 9.8, 4.2, 3.1, 2.8, 1.7),
  labels =
    c( "Sales", "Marketing", "Research",
        "Manufacturing", "Technical Support" ),

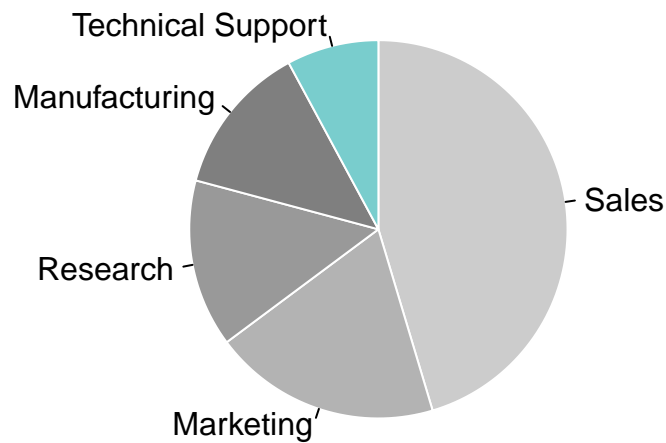
  main = "Distribution of costs for 2019 budget",

  col =
    c( "gray80", "gray70", "gray60",
        "gray50", "darkslategray3"),

  clockwise = TRUE,

  border = "white"
)
```

Distribution of costs for 2019 budget



If you really want to emphasize just one slice of the pie, another strategy is to use the same background color for all the other slices, and then a brighter color for the slice of interest.

Here I've made every slice the color `darkslategray4`, except for the slice for Technical Support, which I've emphasized with the bright color `cyan2`:

```
# Example 10: Pie chart with emphasis

pie(
  c( 9.8, 4.2, 3.1, 2.8, 1.7),

  labels =
    c( "Sales", "Marketing", "Research",
        "Manufacturing", "Technical Support" ),

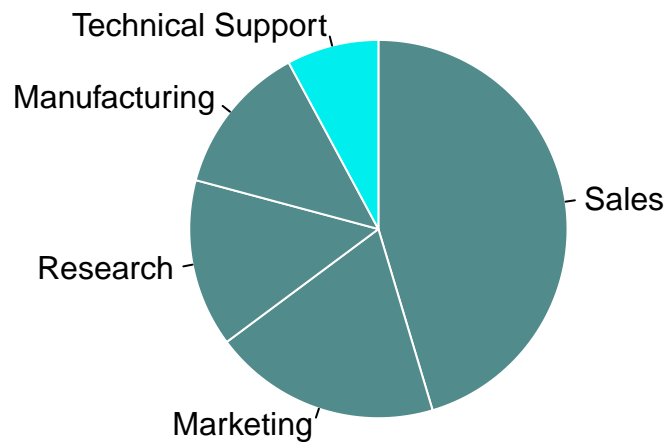
  main = "Distribution of costs for 2019 budget",

  col =
    c( "darkslategray4", "darkslategray4",
        "darkslategray4", "darkslategray4",
        "cyan2" ),

  clockwise = TRUE,

  border = "white"
)
```

Distribution of costs for 2019 budget



So those are some strategies for selecting colors for a pie chart.

Now let's review what we've learned in this module.

Exercise 5.3: Colors for the Babe Ruth pie chart

Now explicitly specify the pie slice colors for the Babe Ruth pie chart.

Solution

Module Review

In this module, we learned what a *pie chart* is, and how to create it.

- In Section 1, we defined the concept of a pie chart, and discuss some of the advantages and disadvantages of this visualization method.
- In Section 2, we constructed a basic pie chart.
- In Section 3, we modified the basic pie chart.
- In Section 4, we explored some strategies for selecting colors for a pie chart.

At the end of this module, you should be able to:

- Create a basic pie chart given a set of numeric values.
- Modify the pie chart by explicitly specifying labels, titles, and colors.

We also met one new built-in R function.

- `pie()`

All right ! That's it for Module 6: Pie Charts.

In fact, that's the end of the content for Week 2: Numeric Values.

Now you can finished Problem Set 2.

Solutions to the Exercises

Exercise 5.1: Babe Ruth At-Bats

Recall that an at-bat is either an out or a hit, and there are four different kinds of hits.

Thus, there are five possible outcomes for an at-bat:

- An out
- A single
- A double
- A triple

Let's make a simple pie chart to display the relative proportions of these outcomes for Babe Ruth.

Recall that Babe Ruth had these batting statistics:

Statistics	Value
Plate appearances	10,626
At-bats	8,399
Hits	2,873
Doubles	506
Triples	136
Home runs	714
Bases on balls	2,062
Hit by a pitch	43
Sacrifice flies	0

Remember these two derived statistics:

- To calculate the number of hits, subtract the number of hits from the number of at-bats
- To calculate the number of singles, subtract the number of doubles, triples, and home runs from the number of hits.

Here are some variables for you:

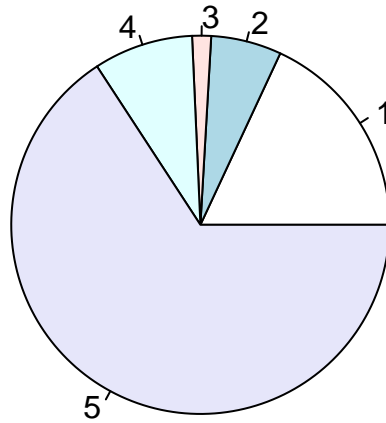
```
babe.ruth.plate.appearances <- 10626
babe.ruth.at.bats <- 8399
babe.ruth.hits <- 2873
babe.ruth.doubles <- 506
babe.ruth.triples <- 136
babe.ruth.home.runs <- 714
```

Solution

```
babe.ruth.outs <-
  babe.ruth.at.bats - babe.ruth.hits

babe.ruth.singles <-
  babe.ruth.hits -
  (babe.ruth.doubles +
   babe.ruth.triples +
   babe.ruth.home.runs)

pie(
  x =
    c(
      babe.ruth.singles,
      babe.ruth.doubles,
      babe.ruth.triples,
      babe.ruth.home.runs,
      babe.ruth.outs
    )
)
```



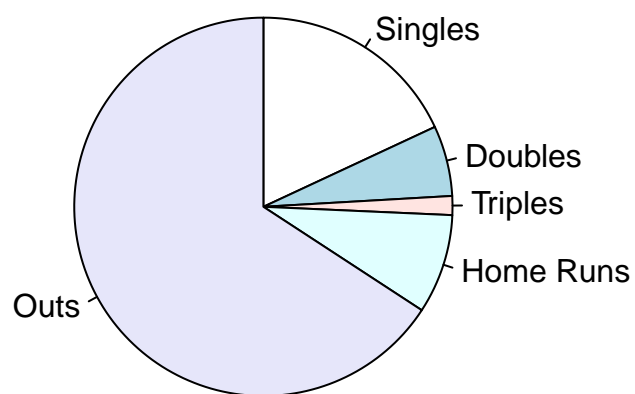
Exercise 5.2: Modifying the Babe Ruth pie chart

Now add labels and a main title to the Babe Ruth pie chart, and set the rotation to clockwise.

Solution

```
pie(  
  x =  
    c(  
      babe.ruth.singles,  
      babe.ruth.doubles,  
      babe.ruth.triples,  
      babe.ruth.home.runs,  
      babe.ruth.outs  
    ),  
  
  labels =  
    c( "Singles", "Doubles", "Triples",  
       "Home Runs", "Outs"),  
  
  main = "Babe Ruth Career At-Bats",  
  
  clockwise = TRUE  
)
```

Babe Ruth Career At-Bats



Exercise 5.3: Colors for the Babe Ruth pie chart

Now explicitly specify the pie slice colors for the Babe Ruth pie chart.

Solution

```
pie(  
  x =  
    c(  
      babe.ruth.singles,  
      babe.ruth.doubles,  
      babe.ruth.triples,  
      babe.ruth.home.runs,  
      babe.ruth.outs  
    ),  
  
  labels =  
    c( "Singles", "Doubles", "Triples",  
       "Home Runs", "Outs"),  
  
  main = "Babe Ruth Career At-Bats",  
  
  clockwise = TRUE,  
  
  col =
```



```
c( "cyan1", "cadetblue2", "darkslategray3",  
  "royalblue", "salmon1" )  
)
```

Babe Ruth Career At-Bats

