

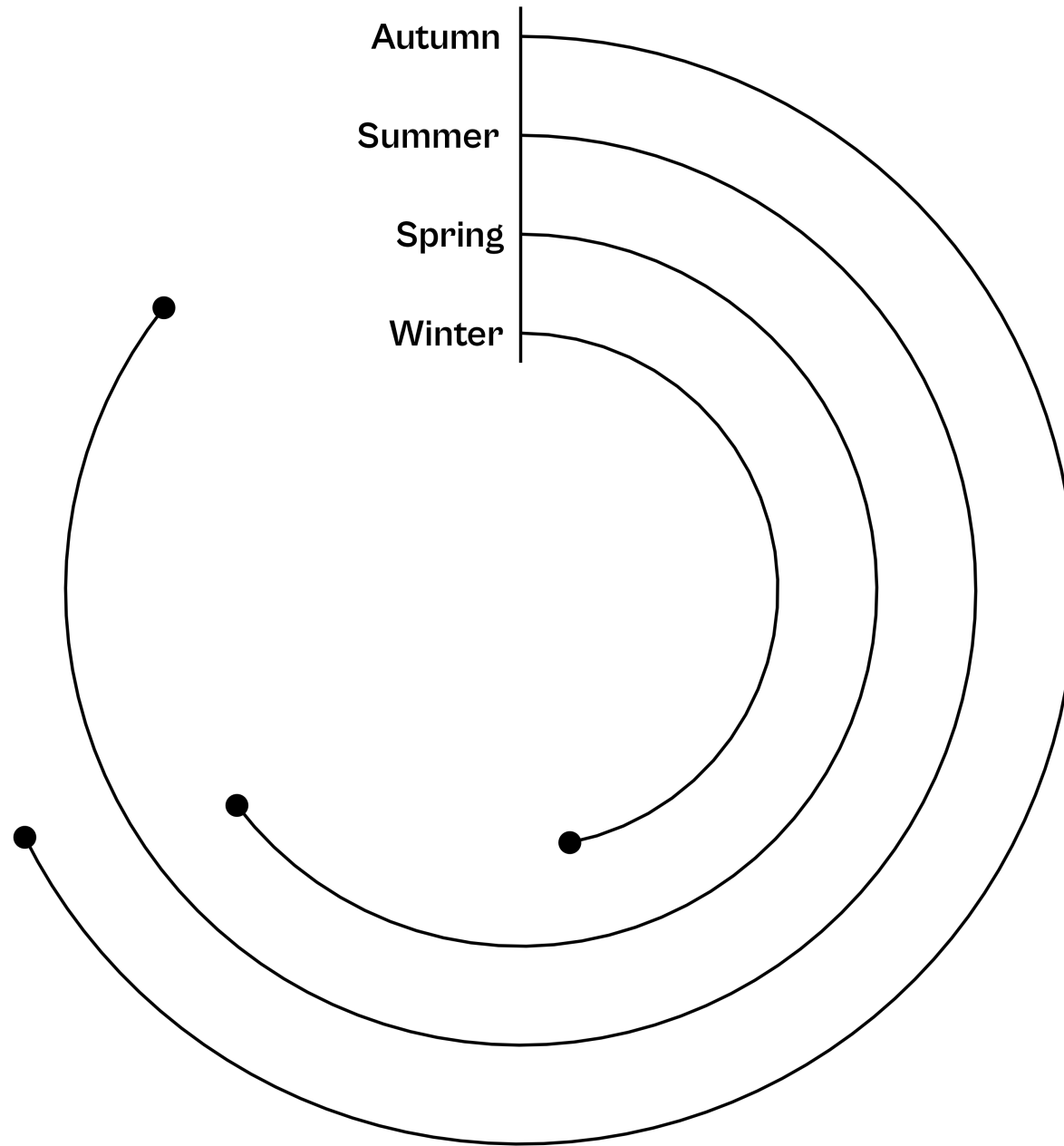
Graphic Design with ggplot2

Concepts of the {ggplot2} Package Pt. 2: Solution Exercise 2

Cédric Scherer // rstudio::conf // July 2022

Exercise 2

- Create a circular lollipop plot of reported bike shares per season.
 - The data is not in the right format as there are no totals.
How can you solve it?
 - Remove all legend elements (with a single line of code).
 - How can you add the labels next to the starting point of each lollipop?
 - How could you add a baseline?



Preparation

```
1 library(tidyverse)
2
3 bikes <- readr::read_csv(
4   "https://raw.githubusercontent.com/z3tt/graphic-design-ggplot2/main/data/london-bikes-custom.csv",
5   col_types = "Dcffffilll1dddc"
6 )
7
8 bikes$season <- forcats::fct_inorder(bikes$season)
9
10 theme_set(theme_light(base_size = 14, base_family = "Roboto Condensed"))
```

Lollipop Plot with Pre-Calculated Sums

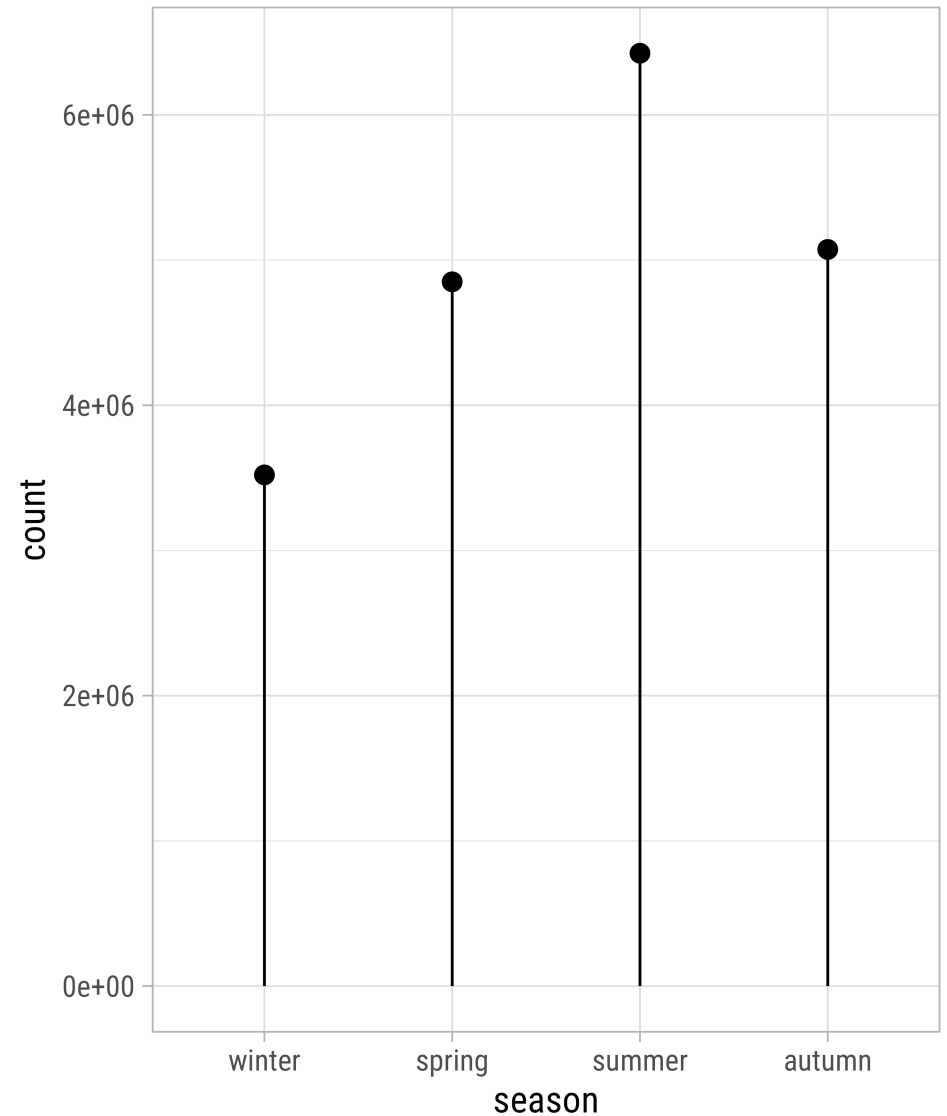
```
1 bikes %>%  
2   group_by(season) %>%  
3   summarize(count = sum(count))
```

```
# A tibble: 4 x 2
```

	season	count
	<fct>	<int>
1	winter	3520407
2	spring	4850236
3	summer	6424609
4	autumn	5073040

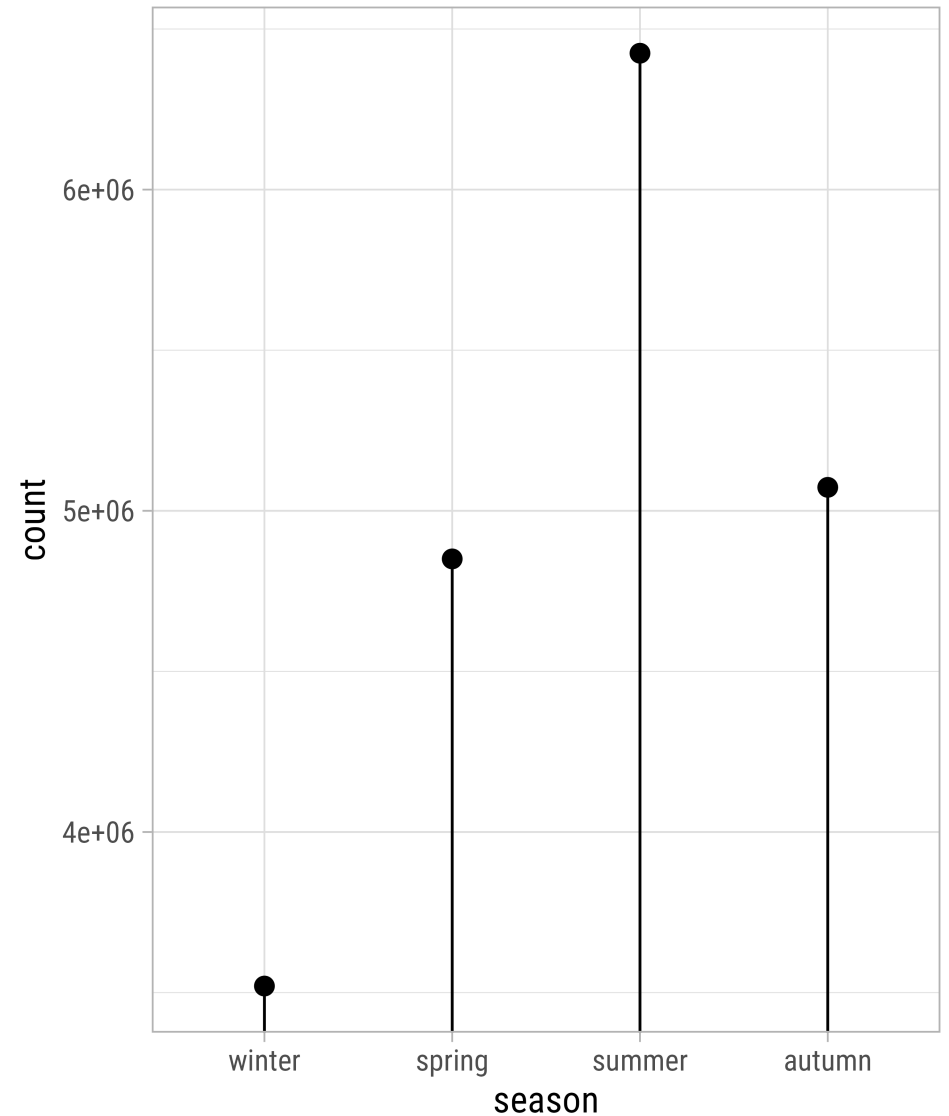
Lollipop Plot with Pre-Calculated Sums

```
1 bikes %>%  
2   group_by(season) %>%  
3   summarize(count = sum(count)) %>%  
4   ggplot(aes(x = season, y = count)) +  
5   geom_point(size = 3) +  
6   geom_linerange(  
7     aes(ymin = 0, ymax = count)  
8   )
```



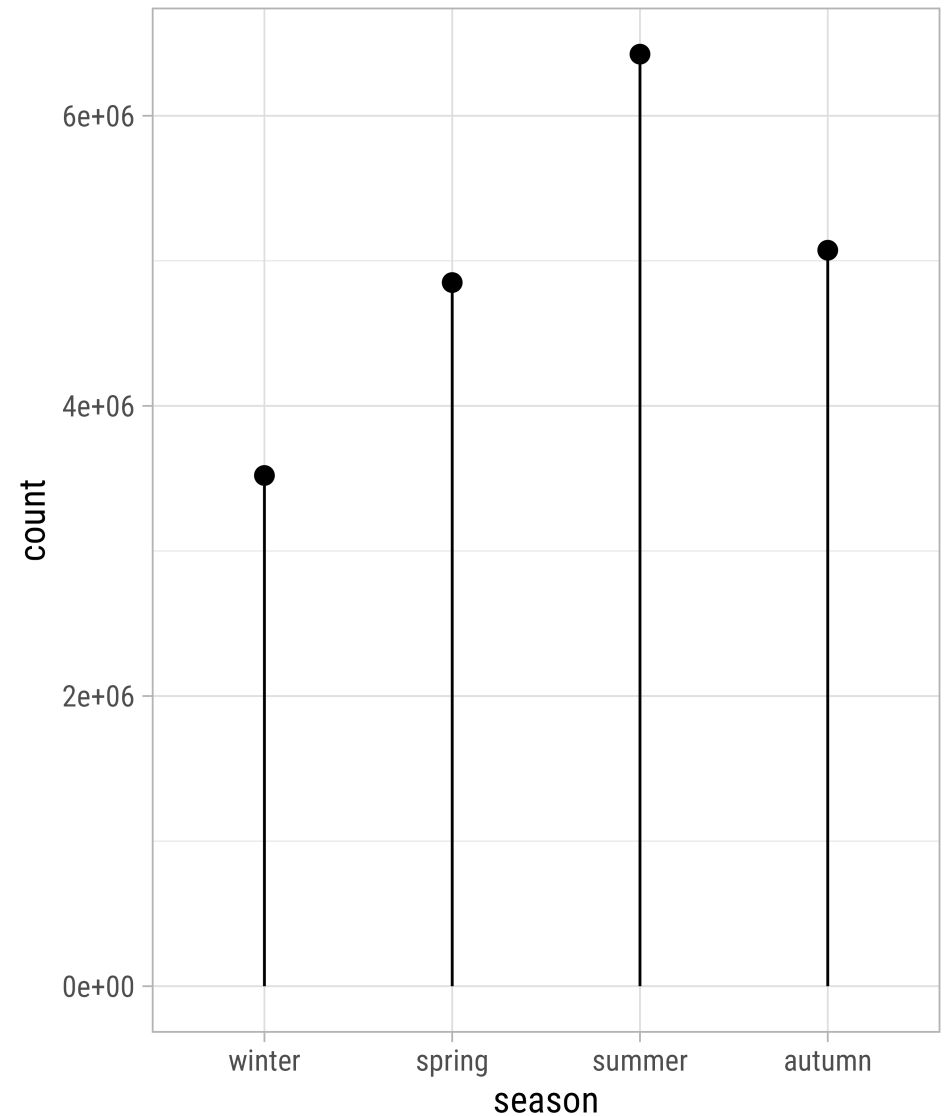
Calculate Sums via `stat_summary()`

```
1 ggplot(bikes, aes(x = season, y = count)) +  
2   stat_summary(  
3     geom = "point", fun = "sum", size = 3  
4   ) +  
5   stat_summary(  
6     geom = "linerange", ymin = 0,  
7     fun.max = function(y) sum(y)  
8   )
```



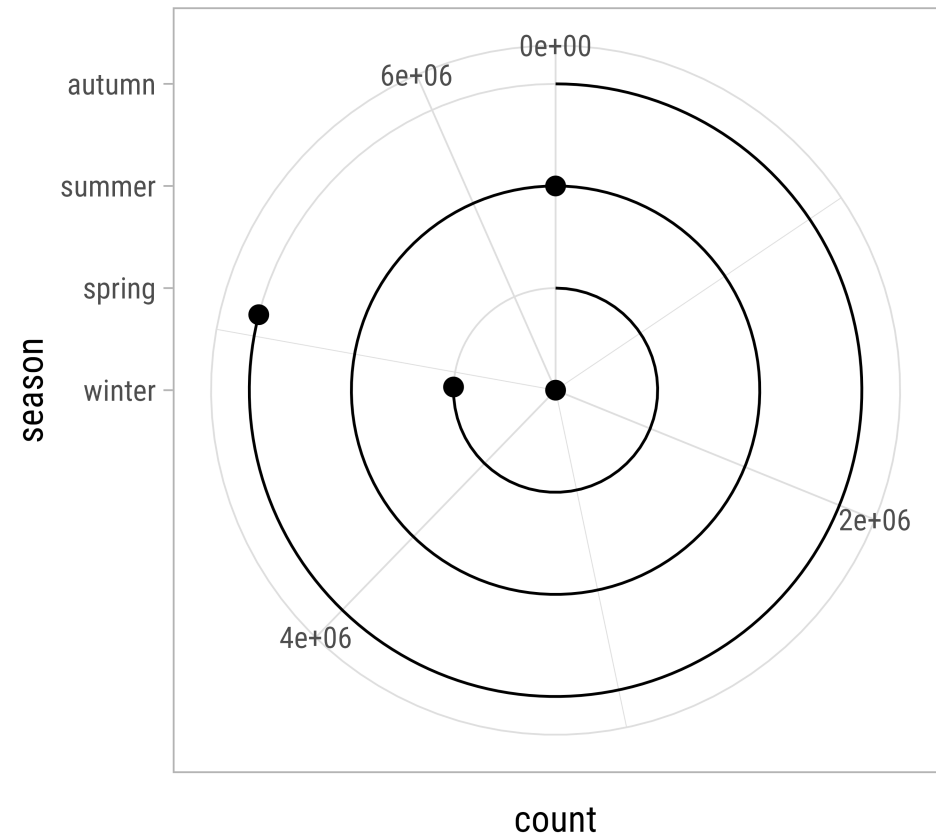
Calculate Sums via `stat_summary()`

```
1 ggplot(bikes, aes(x = season, y = count)) +  
2   stat_summary(  
3     geom = "point", fun = "sum", size = 3  
4   ) +  
5   stat_summary(  
6     geom = "linerange", ymin = 0,  
7     fun.max = function(y) sum(y)  
8   ) +  
9   coord_cartesian(ylim = c(0, NA))
```



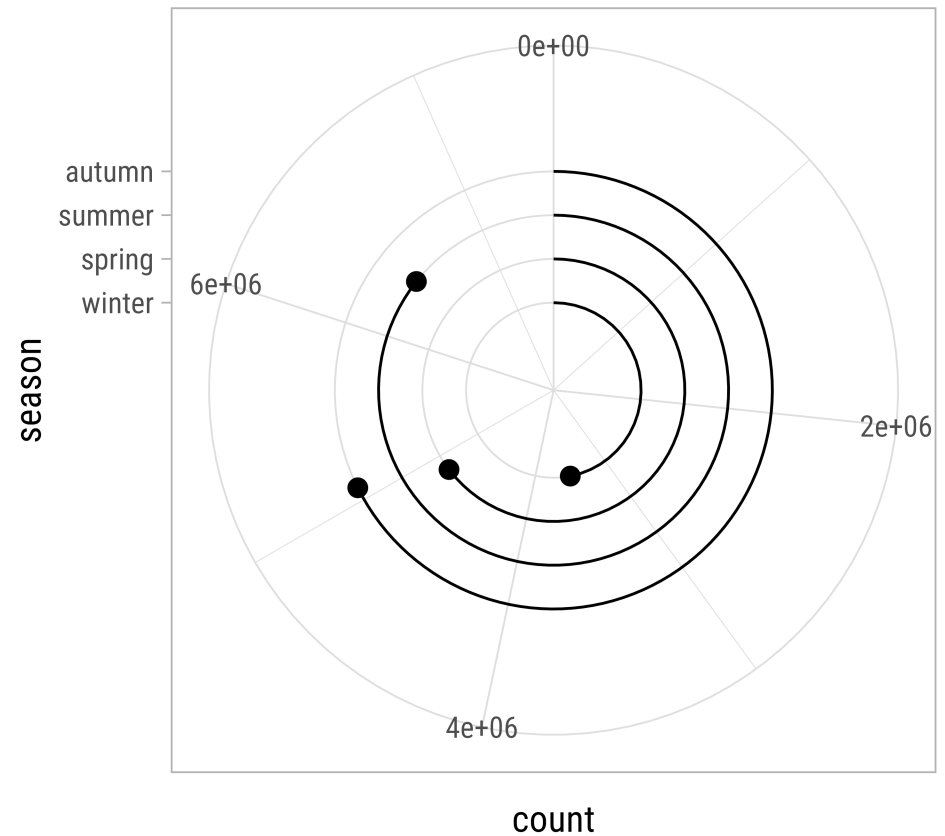
Apply a Polar Coordinate System

```
1 bikes %>%  
2   group_by(season) %>%  
3   summarize(count = sum(count)) %>%  
4   ggplot(aes(x = season, y = count)) +  
5   geom_point(size = 3) +  
6   geom_linerange(  
7     aes(ymin = 0, ymax = count)  
8   ) +  
9   coord_polar(theta = "y")
```



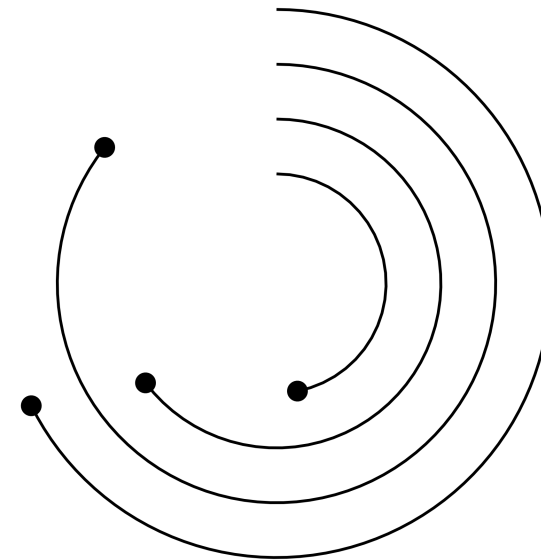
Fix Axis Ranges

```
1 bikes %>%  
2   group_by(season) %>%  
3   summarize(count = sum(count)) %>%  
4   ggplot(aes(x = season, y = count)) +  
5   geom_point(size = 3) +  
6   geom_linerange(  
7     aes(ymin = 0, ymax = count)  
8   ) +  
9   coord_polar(theta = "y") +  
10  scale_x_discrete(  
11    expand = c(.5, .5)  
12  ) +  
13  scale_y_continuous(  
14    limits = c(0, 7.5*10^6)  
15  )
```



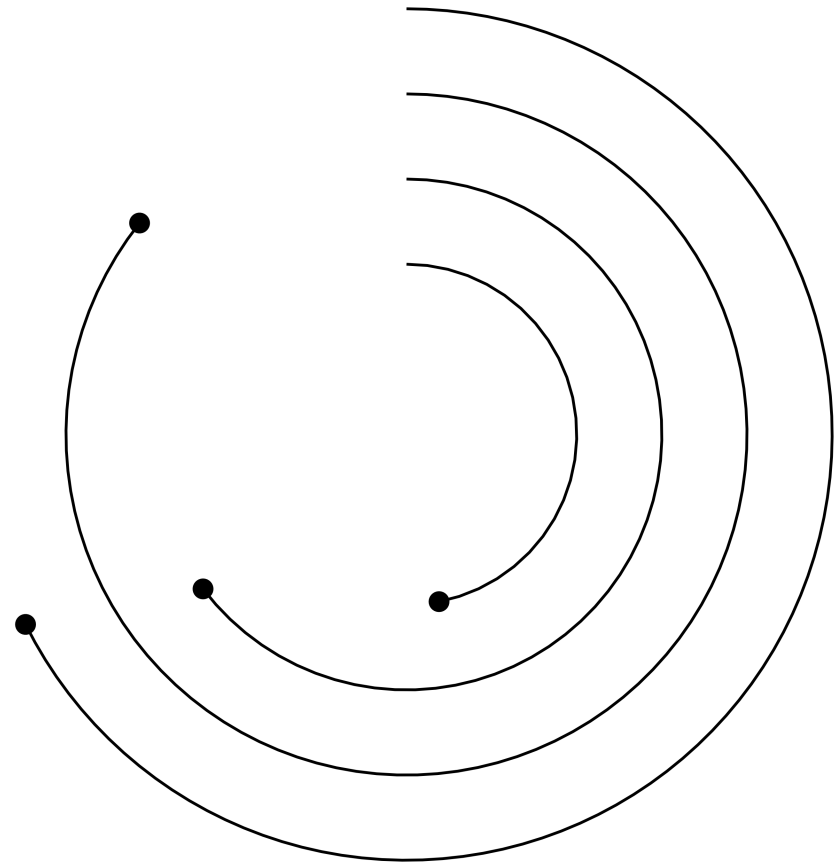
Remove All Theme Components

```
1 bikes %>%
2   group_by(season) %>%
3   summarize(count = sum(count)) %>%
4   ggplot(aes(x = season, y = count)) +
5   geom_point(size = 3) +
6   geom_linerange(
7     aes(ymin = 0, ymax = count)
8   ) +
9   coord_polar(theta = "y") +
10  scale_x_discrete(
11    expand = c(.5, .5)
12  ) +
13  scale_y_continuous(
14    limits = c(0, 7.5*10^6)
15  ) +
16  theme_void()
```



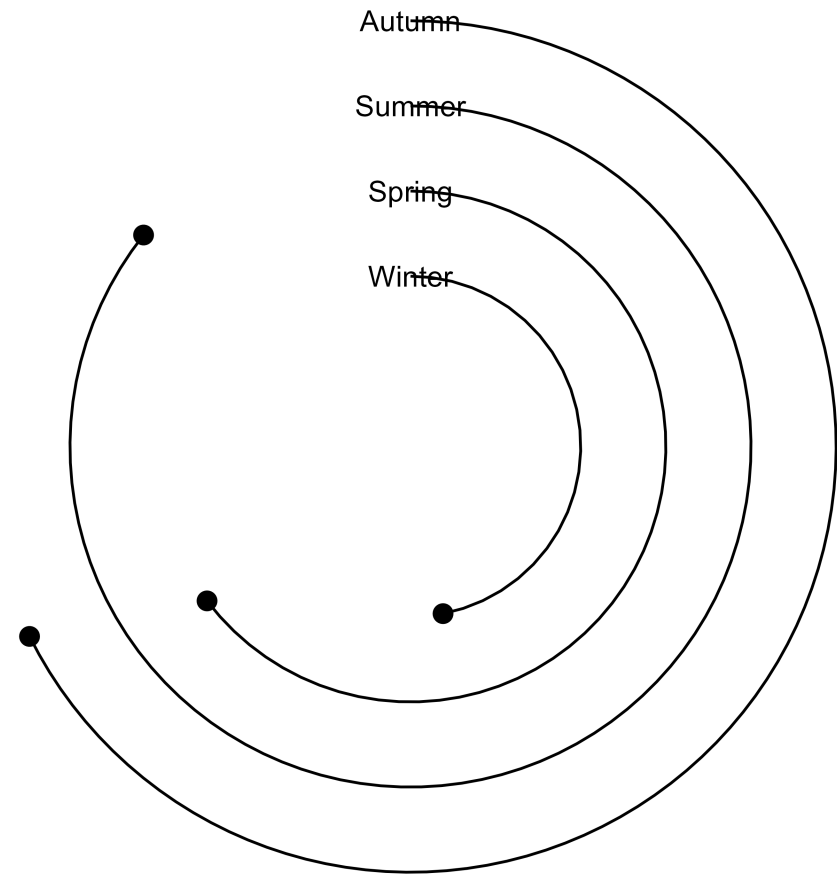
Fix Plot Margin

```
1 bikes %>%
2   group_by(season) %>%
3   summarize(count = sum(count)) %>%
4   ggplot(aes(x = season, y = count)) +
5   geom_point(size = 3) +
6   geom_linerange(
7     aes(ymin = 0, ymax = count)
8   ) +
9   coord_polar(theta = "y") +
10  scale_x_discrete(
11    expand = c(.5, .5)
12  ) +
13  scale_y_continuous(
14    limits = c(0, 7.5*10^6)
15  ) +
16  theme_void() +
17  theme(plot.margin = margin(rep(-100, 4)))
```



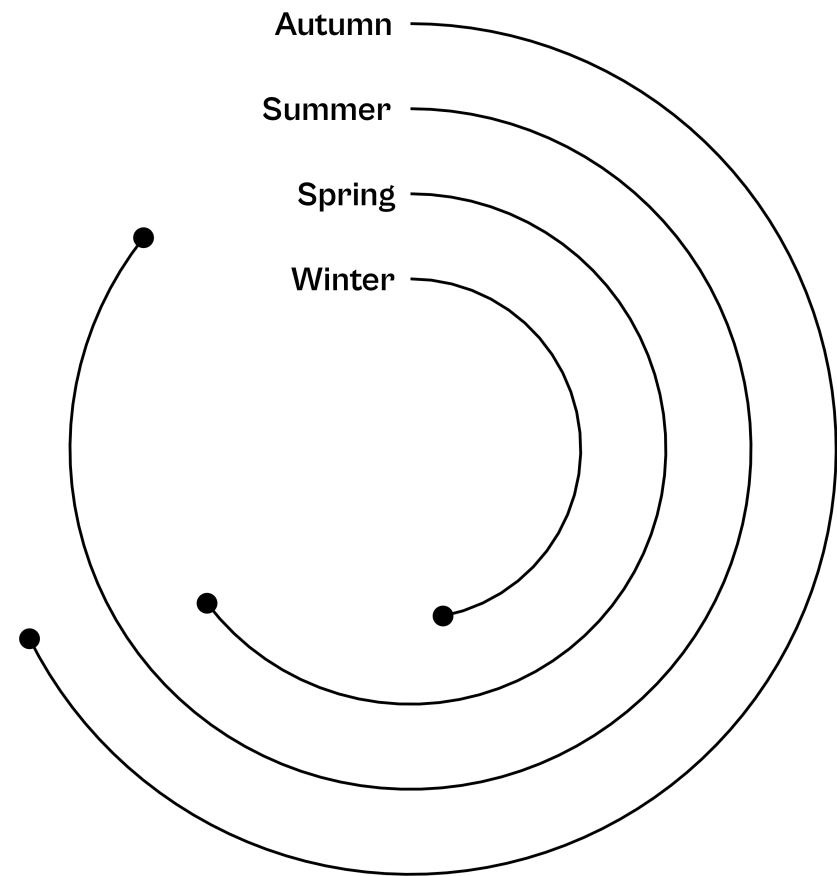
Add “Axis” Text

```
1 bikes %>%
2   group_by(season) %>%
3   summarize(count = sum(count)) %>%
4   ggplot(aes(x = season, y = count)) +
5   geom_point(size = 3) +
6   geom_linerange(
7     aes(ymin = 0, ymax = count)
8   ) +
9   geom_text(
10    aes(label = stringr::str_to_title(season))
11  ) +
12  coord_polar(theta = "y") +
13  scale_x_discrete(
14    expand = c(.5, .5)
15  ) +
16  scale_y_continuous(
17    limits = c(0, 7.5*10^6)
18  ) +
19  theme_void() +
20  theme(plot.margin = margin(rep(-100, 4)))
```



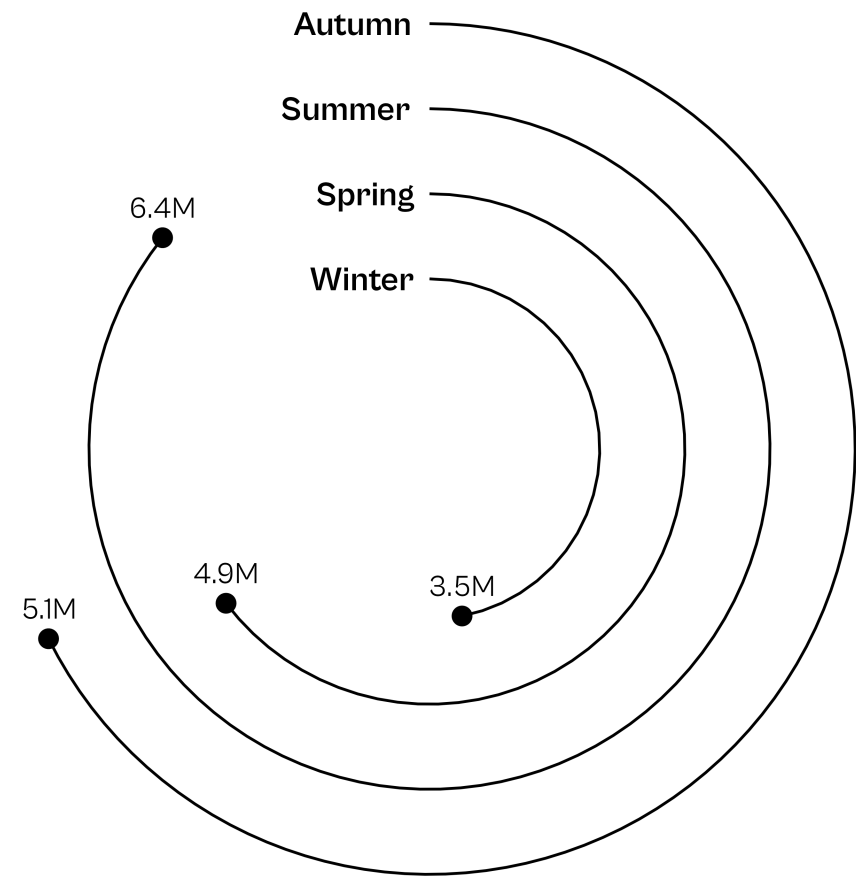
Style “Axis” Text

```
1 bikes %>%
2   group_by(season) %>%
3   summarize(count = sum(count)) %>%
4   ggplot(aes(x = season, y = count)) +
5   geom_point(size = 3) +
6   geom_linerange(
7     aes(ymin = 0, ymax = count)
8   ) +
9   geom_text(
10    aes(label = stringr::str_to_title(season),
11        family = "Cabinet Grotesk", size = 4.5,
12        fontface = "bold", hjust = 1.15
13    ) +
14    coord_polar(theta = "y") +
15    scale_x_discrete(
16      expand = c(.5, .5)
17    ) +
18    scale_y_continuous(
19      limits = c(0, 7.5*10^6)
20    ) +
21    theme_void() +
22    theme(plot.margin = margin(rep(-100, 4)))
```



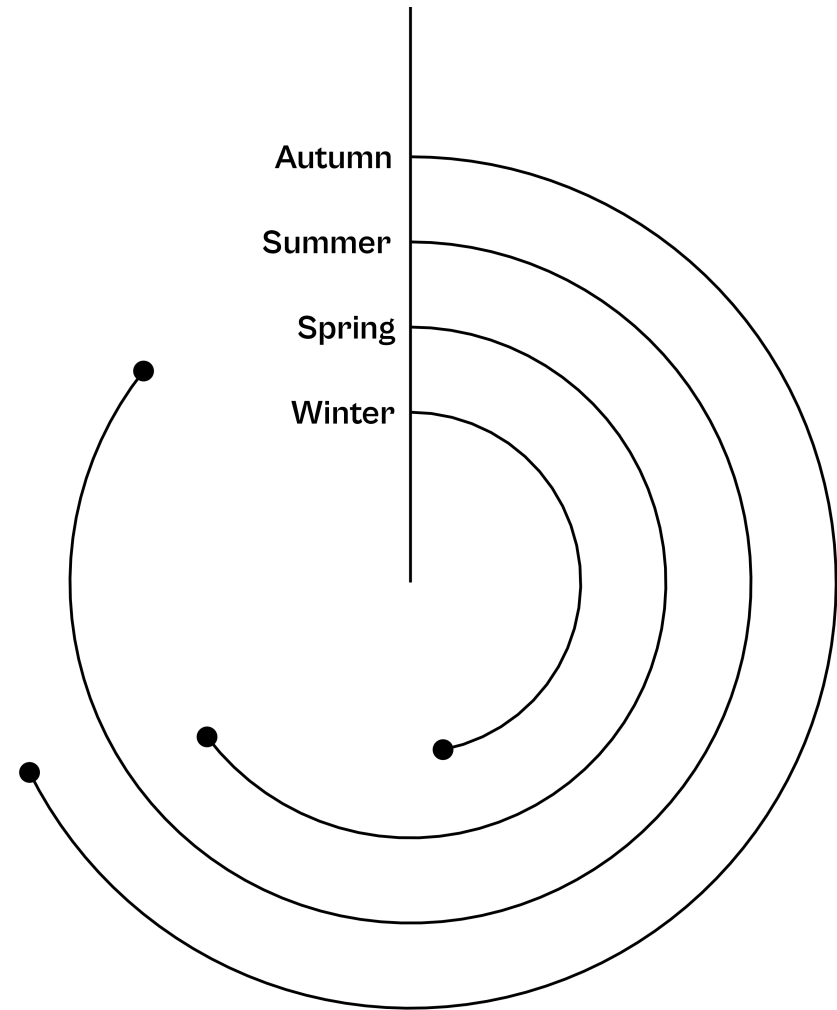
Alternatively: Add Direct Labels

```
1 bikes %>%
2   group_by(season) %>%
3   summarize(count = sum(count)) %>%
4   ggplot(aes(x = season, y = count)) +
5   geom_point(size = 3) +
6   geom_linerange(
7     aes(ymin = 0, ymax = count)
8   ) +
9   geom_text(
10    aes(label = stringr::str_to_title(season),
11        family = "Cabinet Grotesk", size = 4.5,
12        fontface = "bold", hjust = 1.15
13    ) +
14    geom_text(
15      aes(label = paste0(round(count / 10^6, 1), "M"),
16          size = 4, vjust = -1, family = "Cabinet Grotesk"
17    ) +
18    coord_polar(theta = "y") +
19    scale_x_discrete(
20      expand = c(.5, .5)
21    ) +
22    scale_y_continuous(
23      limits = c(0, 7.5*10^6)
```



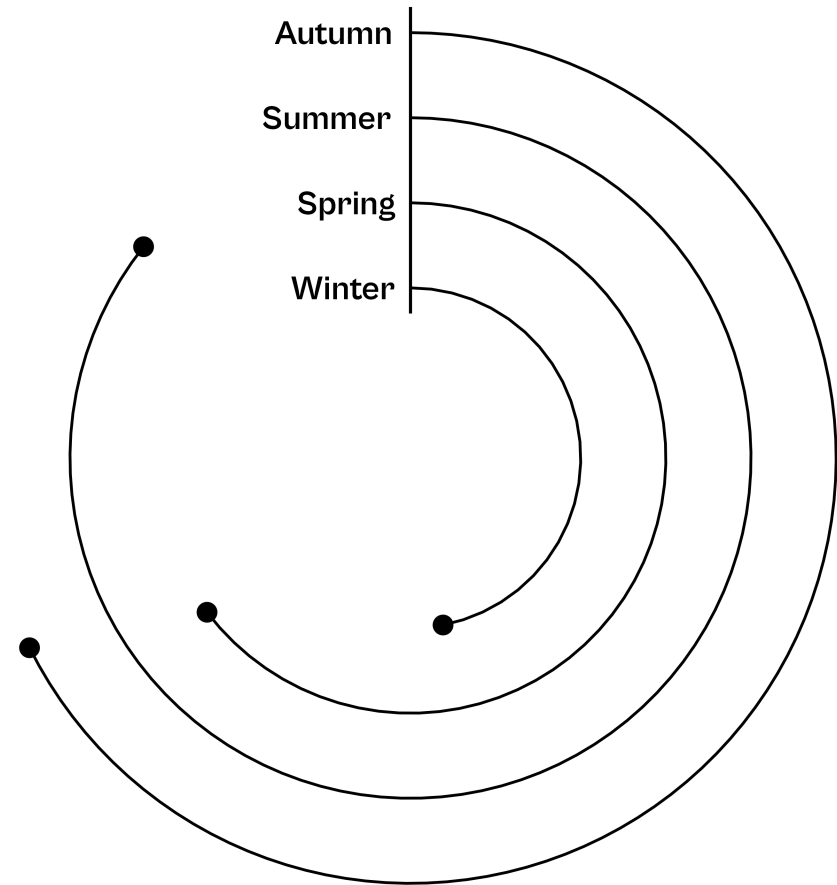
Add a Baseline — *ugly but simple*

```
1 bikes %>%
2   group_by(season) %>%
3   summarize(count = sum(count)) %>%
4   ggplot(aes(x = season, y = count)) +
5   geom_point(size = 3) +
6   geom_linerange(
7     aes(ymin = 0, ymax = count)
8   ) +
9   geom_hline(yintercept = 0) +
10  geom_text(
11    aes(label = stringr::str_to_title(season),
12        family = "Cabinet Grotesk", size = 4.5,
13        fontface = "bold", hjust = 1.15
14    ) +
15    coord_polar(theta = "y") +
16    scale_x_discrete(
17      expand = c(.5, .5)
18    ) +
19    scale_y_continuous(
20      limits = c(0, 7.5*10^6)
21    ) +
22    theme_void() +
23    theme(plot.margin = margin(rep(-100, 4)))
```



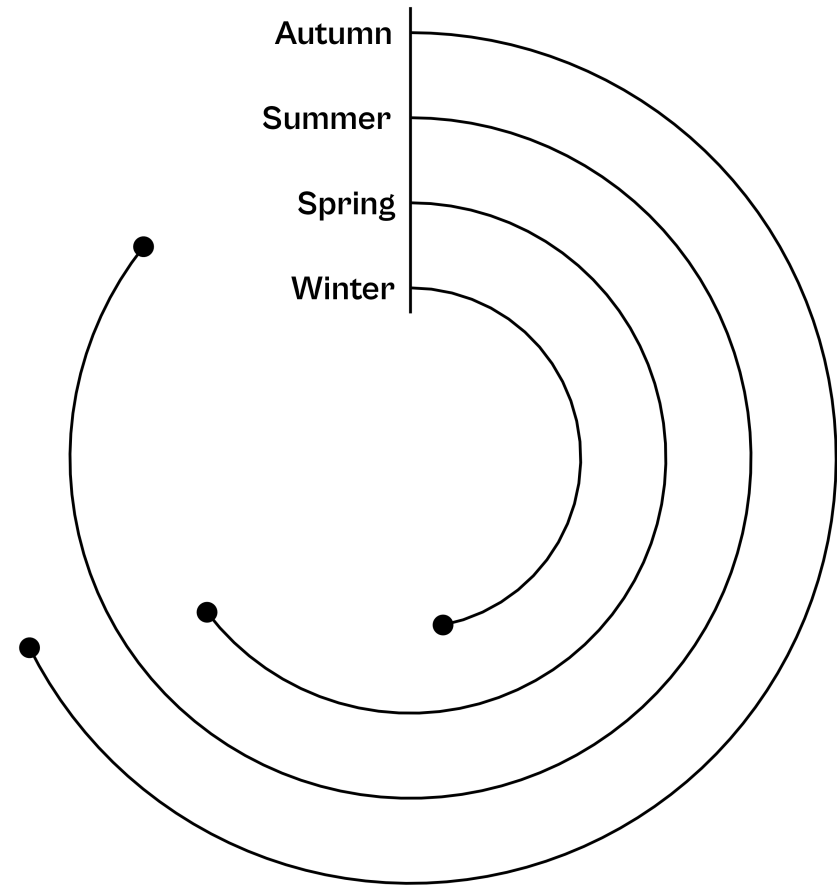
Add a Baseline — *nice but unusual*

```
1 bikes %>%
2   group_by(season) %>%
3   summarize(count = sum(count)) %>%
4   ggplot(aes(x = season, y = count)) +
5   geom_point(size = 3) +
6   geom_linerange(
7     aes(ymin = 0, ymax = count)
8   ) +
9   geom_linerange(
10    xmin = .7, xmax = 4.3, y = 0
11  ) +
12  geom_text(
13    aes(label = stringr::str_to_title(season),
14        family = "Cabinet Grotesk", size = 4.5,
15        fontface = "bold", hjust = 1.15
16  ) +
17  coord_polar(theta = "y") +
18  scale_x_discrete(
19    expand = c(.5, .5)
20  ) +
21  scale_y_continuous(
22    limits = c(0, 7.5*10^6)
23  ) +
```



Add a Baseline — *yeah, that's it!*

```
1 bikes %>%
2   group_by(season) %>%
3   summarize(count = sum(count)) %>%
4   ggplot(aes(x = season, y = count)) +
5   geom_point(size = 3) +
6   geom_linerange(
7     aes(ymin = 0, ymax = count)
8   ) +
9   annotate(
10     geom = "linrange",
11     xmin = .7, xmax = 4.3, y = 0
12   ) +
13   geom_text(
14     aes(label = stringr::str_to_title(season),
15         family = "Cabinet Grotesk", size = 4.5,
16         fontface = "bold", hjust = 1.15
17     ) +
18     coord_polar(theta = "y") +
19     scale_x_discrete(
20       expand = c(.5, .5)
21     ) +
22     scale_y_continuous(
23       limits = c(0, 7.5*10^6)
```



Solution using `stat_summary()`

```
1 ggplot(bikes, aes(x = as.numeric(season), y =  
2   stat_summary(  
3     geom = "point", fun = "sum", size = 3  
4   ) +  
5   stat_summary(  
6     geom = "linerange", ymin = 0,  
7     fun.max = function(y) sum(y)  
8   ) +  
9   stat_summary(  
10    geom = "text",  
11    aes(  
12      label = stringr::str_to_title(season),  
13      y = 0  
14    ),  
15    family = "Cabinet Grotesk", size = 4.5,  
16    fontface = "bold", hjust = 1.15  
17  ) +  
18  annotate(  
19    geom = "linerange",  
20    xmin = .7, xmax = 4.3, y = 0  
21  ) +  
22  coord_polar(theta = "y") +  
23  scale_x_discrete(
```

