Creating a map

The default projection is the Mercator projection

```
library(maps)
states_map <- map_data("state")
ggplot(states_map,
  aes(x=long,y=lat,group=group)) +
geom_polygon(fill="white",colour="black")

ggplot(states_map, aes(x=long,y=lat,group=group)) + geom_
```

The `map data()` function

long Longtitude

lat Latitude

group This is a grouping variable for each polygon. A region or subregion might have multiple polygons.

order The order to connect each point in a group

region Roughly, the names of countries, although some other objects are lakes.

subregion The names of subregions with a region, which can be obtained for multiple groups

There are a number of different maps available, including `world`, `nz`, `france`,`italy`,`usa` (outline of USA),`state` (each state in the USA) and `county` (each county in the USA).

To get a map of the world

```
world_map <- map_data("world")
head(world_map)
```

```
       long      lat group order region subregion
1 -133.3664 58.42416     1     1 Canada      <NA>
2 -132.2681 57.16308     1     2 Canada      <NA>
3 -132.0498 56.98610     1     3 Canada      <NA>
4 -131.8797 56.74001     1     4 Canada      <NA>
5 -130.2492 56.09945     1     5 Canada      <NA>
6 -130.0131 55.91169     1     6 Canada      <NA>
```

Maps with ggplot2

If you want to draw a map of a region in the world map for which there isn't a separate map, you can first look for the region name

```
sort(unique(world_map$region))

#East Asia Example in Book

east_asia <- map_data("world",region=c("Japan","China",
"North Korea","South Korea")).
```

Maps with ggplot2

```
ggplot(east_asia, aes(x=long,y=lat,
   group=group,fill=region)) +
   geom_polygon(colours="black")+
   scale_fill_brewer(palette="Set2")
```

It is possible to get speciific regions from a particular map.

```
east_asia <- map_data("world",
    region=c("japan","China","North Korea","South Korea")

#Map regions to fill colour

scale_fill_brewer("pallette_set2")
```

If there is a separate map for each region, that map data will be at higher resolution that in youw ere to extract it from the "world" map.

```
NZ1 <- map_data("world",region="New Zealand")
NZ1 < subset(NZ1, long>0 &lat> -48)  #trim off outlying i
ggplot(NZ1,aes(x=long,y=lat,group=group)) + geom_path()

NZ2 <- map_data("nz")
ggplot(nz2 (aes(x=long,y=lat)) + geom_path()
```

Chloropleth map

Creating a map where regions are colours according to some variable value.

The Crime Data (USArrests)

```
crimes <-data.frame(state =tolower(rownames(USArrests)),U
head(crimes)
```

Chloropleth map

```
library(maps)
states_map <- map_data("state")
crime_map <- merge(states_map,crimes,by.x="region",by.y="
head(crime_map)
```

After merging, the order has changed, leading to polygons drawn in the wrong order. To solve this, the data must be sorted.

```
libary(plyr)
# Use this package for the arrange function().
crime_map <- arrange(crime_map,group,order)
head(crime_map)
```

Chloropleth Maps

```
ggplot(crimes, aes(map_id=state,fill=Assault))+
geom_polygon(colour="black")+
expand_limits(x=states_map$long,y=states_map$lat) +
coord_map("polyconic")
```

Chloropleth Maps

```
ggplot(crimes(aes(map_id=state,fill=Assault))+
geom_map(map=state_map,colour="black") +
scale_fill_gradient2
   (low="#559999",mid="grey90",high="#BB650B",midpoint=me
expand_limits(x=state_map$long,y=state_map$lat) +
co_ordmap("poyconic")
```

Remark The previous example mapped continuous variables to `fill`, but discrete values could have also have been used. It is often easier to interpret the data if the data has been discretized. For example, we could categorize the values into quantiles and show thos quantiles. (Next Example)

```
ggplot(crimes(aes(map_id=state,fill=Assault))+
geom_map(map=state_map,colour="black") +
expand_limits(x=state_map$long,y=state_map$lat) +
co_ordmap("polyconic")
```

Themes

Making a map with a clean background : removing background elements from a map.

```
#Create a theme with many of the
# background elements removed.
theme_clean <-function(base_size=12){
   require(grid)
   there_grey(base_size) %+replace%
       theme(
         axis.title =element_blank(),
         axis.text =element_blank(),
         panel.background =element_blank(),
         panel.grid =element_blank(),
          .......
         )
        }
```

Themes

```
ggplot(crimes(aes(map_id=state,fill=assault_q))+
geom_map(map=state_map,colour="black") +
scale_fill_manual(values=pal)+
expand_limits(x=state_map$long,y=state_map$lat) +
co_ordmap("poyconic") +
labs(fill="Assault Rate \n Percentile")+
theme_clean()
```