BSA - Student Faculty Seminar

Kristin Baltrusaitis

11/29/2018

Introduction to ggplot2

ggplot2:

- based on Leland Wilkinson's Grammar of Graphics
- formal structured perspective on how to describe graphics
- references:

1. R Graphics Cookbook by Winston Chang
2. ggplot2: Elegant Graphics for Data Analysis by Hadley Wickham

Introduction to ggplot2

Basic Structure:

1. Start with ggplot object
2. add components with $+$
3. print

Terminology

- data: What we want to visualize
- geom_: geometric objects that are drawn to represent the data [geom_bar, geom_line, ect]
- aes: aesthetic attributes, the visual properties of geoms [x, y, line color, point shapes, ect]
- scales: control the mapping of data values to aesthetics
- guides: show the viewer how to interpret the visual representation [tick marks, axis labels, ect]

Data

Know thy data..

- The structure of the data (long vs. wide) will play a role in how you build the ggplot objects.
- The format (continuous, categorical, time, ordinal) of variables will play a role in the type of components that you can add to your ggplot object.

```
##      scale per.fny.reports per.athena.reports fny.cdc.cor a
## 1 regional        9.915429          10.079645   0.7551514
## 2 regional        7.251376           5.882256   0.6558362
## 3 regional       11.259143          15.224074   0.7363470
## 4 regional       11.831616          22.421978   0.8028679
```

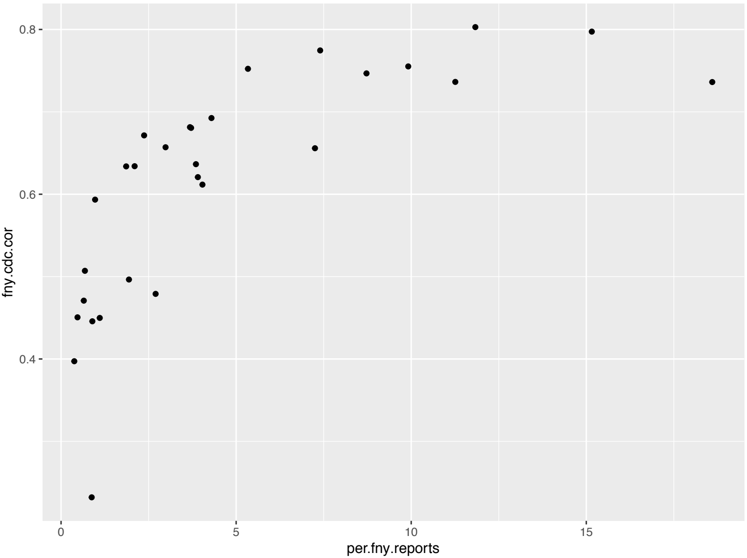
Step 1: create scatterplot

object

```
p1<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor))
```

component

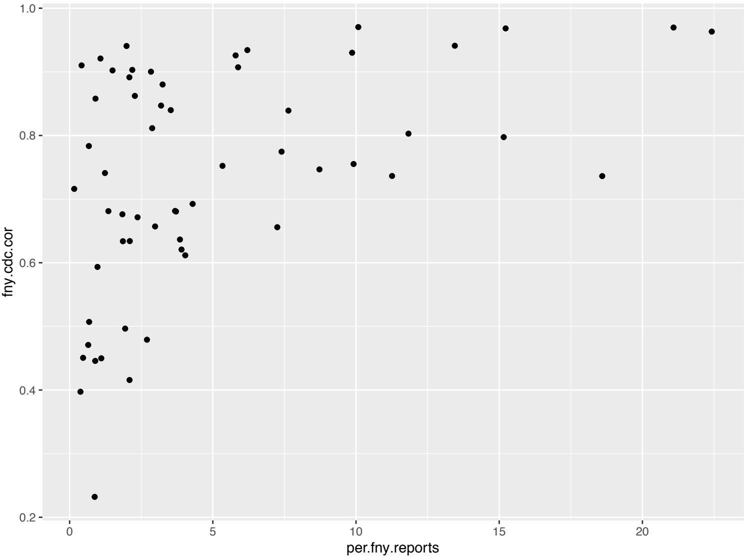
Step 1: create scatterplot



Step 2: add 2nd variable

```
p2<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor))+
  geom_point(aes(x=per.athena.reports, y=athena.cdc.cor))
```

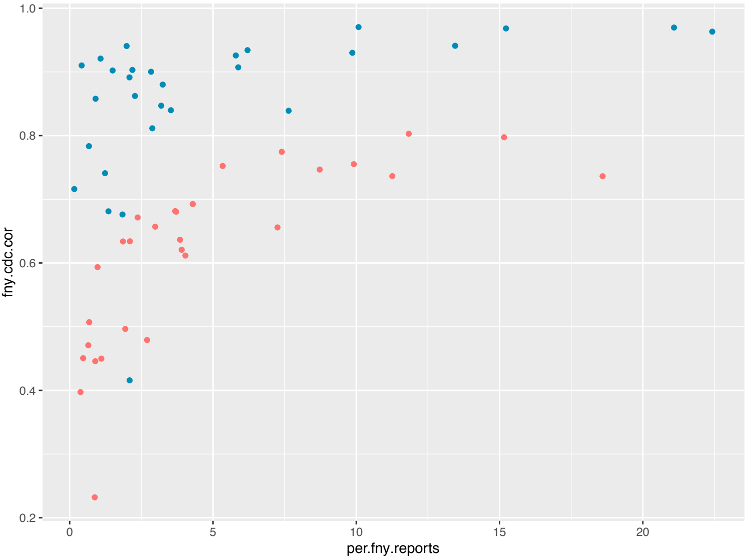
Step 2: add 2nd variable



Step 3: distinguish colors

```
p3<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor),
             color="#FF7270")+
  geom_point(aes(x=per.athena.reports, y=athena.cdc.cor),
             color="#008CB7")
```

Step 3: distinguish colors

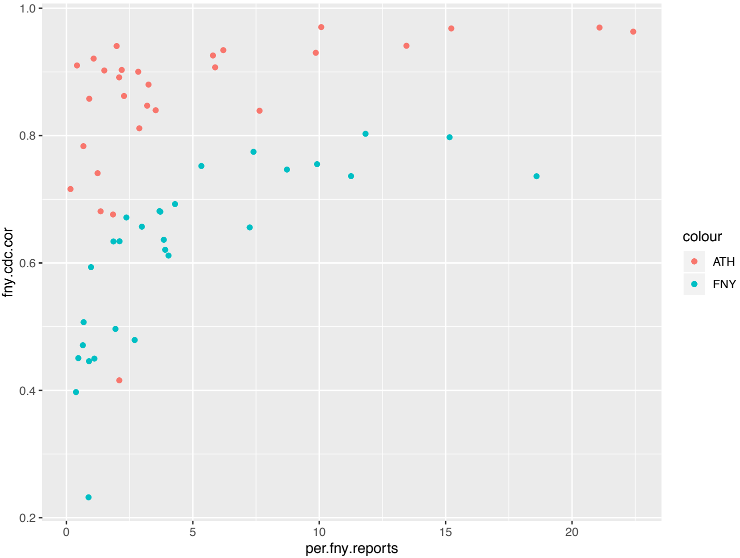


Step 4: add legend

within aes

```
p4<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor,
                 color="FNY"))+
  geom_point(aes(x=per.athena.reports, y=athena.cdc.cor,
                 color="ATH"))
```

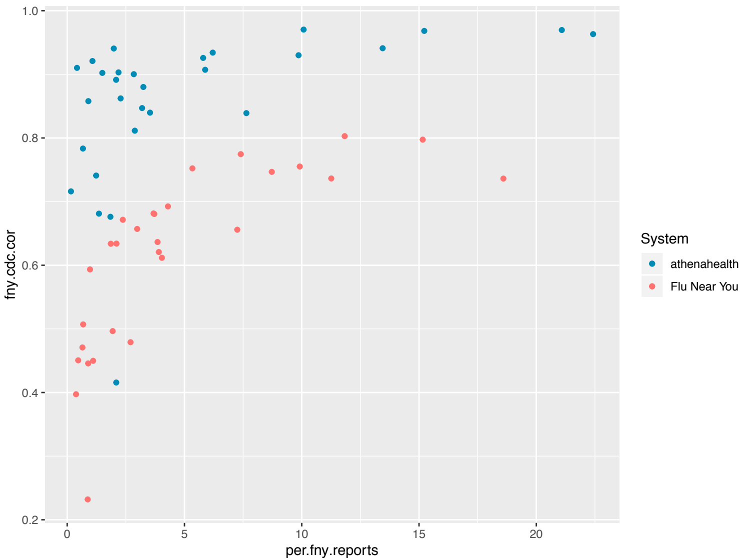
Step 4: add legend



Step 5: add legend and specify colors

```r
p5<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor,
                 color="FNY"))+
  geom_point(aes(x=per.athena.reports,y=athena.cdc.cor,
                 color="ATH"))+
  scale_color_manual(name="System",
    labels = c("athenahealth", "Flu Near You"),
    values=c(ATH="#008CB7", FNY="#FF7270"))
```

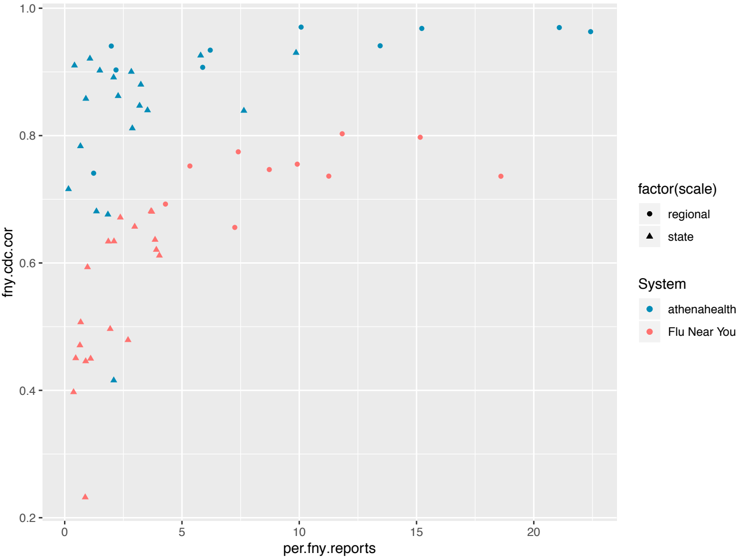
Step 5: add legend and specify colors



Step 6: add shapes to distinguish geographical resolutions

```
p6<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor,
                 color="FNY", shape = factor(scale)))+
  geom_point(aes(x=per.athena.reports, y=athena.cdc.cor,
                 color="ATH", shape = factor(scale)))+
  scale_color_manual(name="System", labels = c("athenahealth"
                     values=c(ATH="#008CB7", FNY="#FF7270"))
```

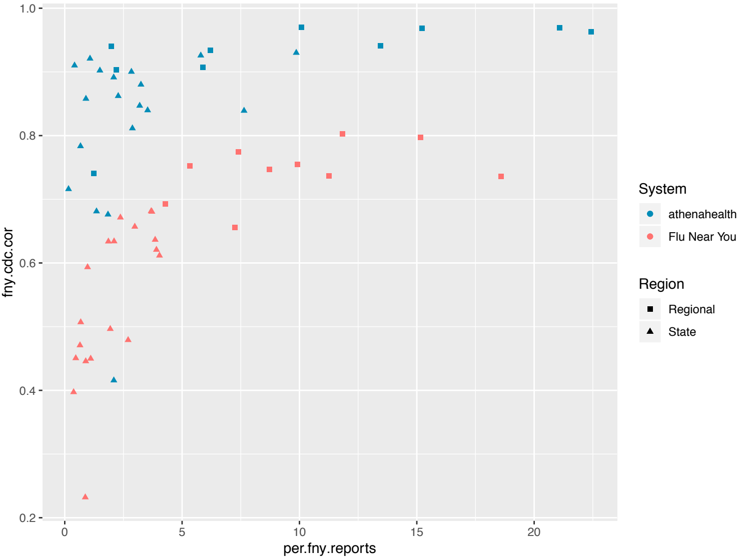
Step 6: add shapes to distinguish geographical resolutions



Step 7: change shapes

```r
p7<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor,
                 color="FNY", shape = factor(scale)))+
  geom_point(aes(x=per.athena.reports, y=athena.cdc.cor,
                 color="ATH", shape = factor(scale)))+
  scale_color_manual(name="System", labels = c("athenahealth"
                     values=c(ATH="#008CB7", FNY="#FF7270"))+
  scale_shape_manual(name="Region",labels=c("Regional", "State
                     values = c(15, 17))
```

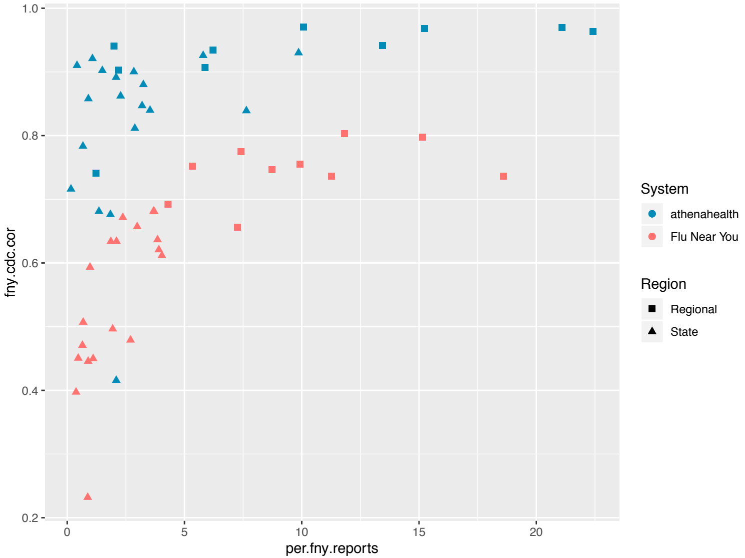
Step 7: change shapes



Step 8: change size

```
p8<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor,
        color="FNY", shape = factor(scale)), size=2)+
  geom_point(aes(x=per.athena.reports, y=athena.cdc.cor,
        color="ATH", shape = factor(scale)), size=2)+
  scale_color_manual(name="System",labels = c("athenahealth",
              values=c(ATH="#008CB7", FNY="#FF7270"))+
  scale_shape_manual(name="Region",labels=c("Regional", "State
                        values = c(15, 17))
```

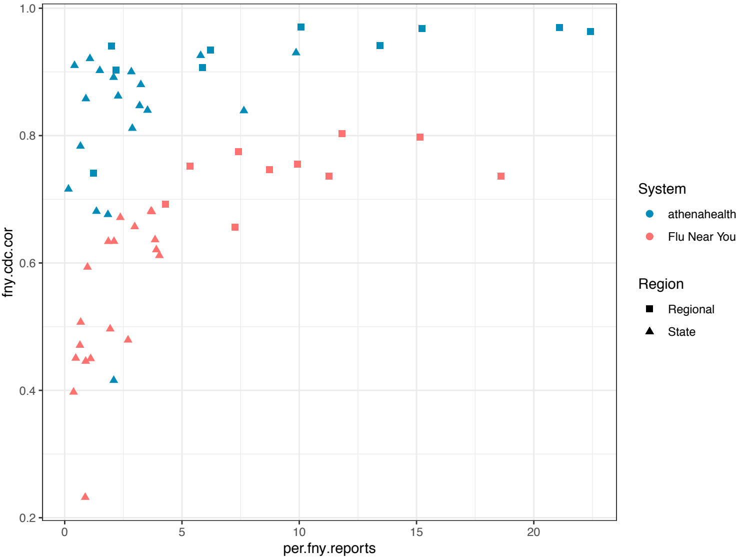
Step 8: change size



Step 9: remove grey background

```
p9<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor,
      color="FNY", shape = factor(scale)), size=2)+
  geom_point(aes(x=per.athena.reports, y=athena.cdc.cor,
        color="ATH", shape = factor(scale)), size=2)+
  theme_bw()+
  scale_color_manual(name="System",labels = c("athenahealth",
              values=c(ATH="#008CB7", FNY="#FF7270"))+
  scale_shape_manual(name="Region",labels=c("Regional", "State
                     values = c(15, 17))
```

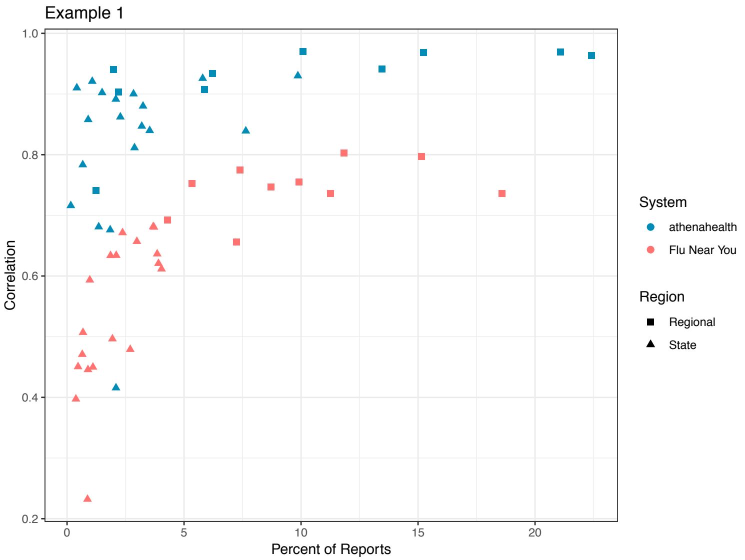
Step 9: remove grey background



Step 10: add axis labels and title

```
p10<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor,
      color="FNY", shape = factor(scale)), size=2)+
  geom_point(aes(x=per.athena.reports, y=athena.cdc.cor,
      color="ATH", shape = factor(scale)), size=2)+
  theme_bw()+
  scale_color_manual(name="System",labels = c("athenahealth",
          values=c(ATH="#008CB7", FNY="#FF7270"))+
  scale_shape_manual(name="Region",labels=c("Regional", "State
              values = c(15, 17))+
  ylab("Correlation")+xlab("Percent of Reports")+
  ggtitle("Example 1")
```

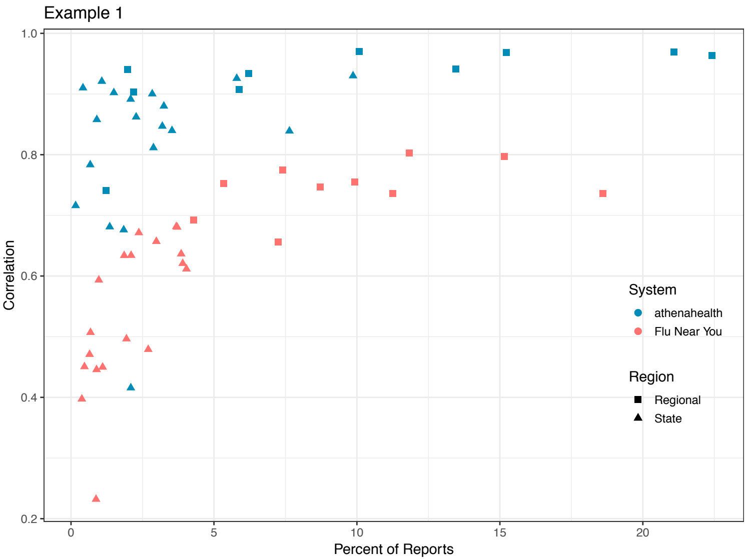
Step 10: add axis labels and title



Step 11: adjust legend

```
p11<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor, color="FNY'
  geom_point(aes(x=per.athena.reports, y=athena.cdc.cor, colo
  theme_bw()+
  scale_color_manual(name="System", labels = c("athenahealth"
  scale_shape_manual(name="Region", labels=c("Regional", "Stat
  ylab("Correlation")+xlab("Percent of Reports")+
  ggtitle("Example 1") +
  theme(legend.justification=c(1,1),
        legend.position=c(0.98,0.5),
        legend.key = element_rect(fill="transparent"),
        legend.background = element_rect(fill="transparent"),
        legend.key.size = unit(0.5, "cm"))
```

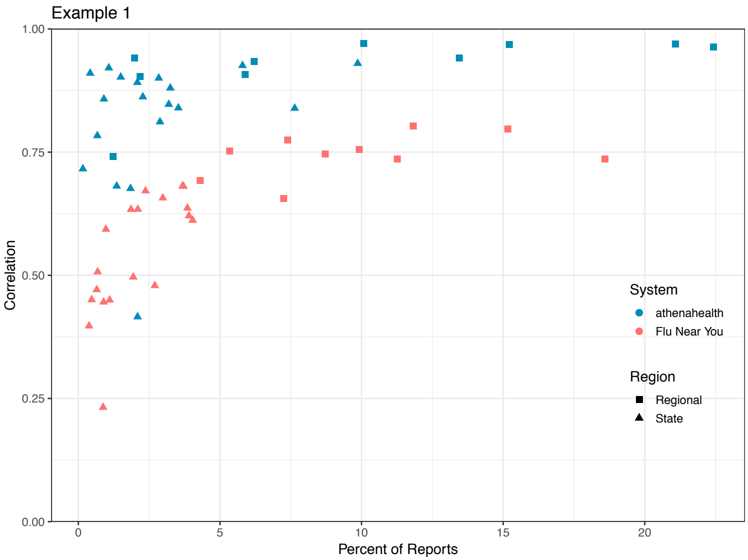
Step 11: adjust legend



Step 12: adjust y-axis

```
p12<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor, color="FNY
  geom_point(aes(x=per.athena.reports, y=athena.cdc.cor, color
  theme_bw()+
  scale_color_manual(name="System",labels = c("athenahealth",
                     values=c(ATH="#008CB7", FNY="#FF7270"))+
  scale_shape_manual(name="Region",labels=c("Regional", "State
  scale_y_continuous( limits = c(0,1), expand = c(0,0) )+
  ylab("Correlation")+xlab("Percent of Reports")+
  ggtitle("Example 1") +
  theme(legend.justification=c(1,1),
        legend.position=c(0.98,0.5),
        legend.key = element_rect(fill="transparent"),
        legend.background = element_rect(fill="transparent"),
        legend.key.size = unit(0.5, "cm"))
```

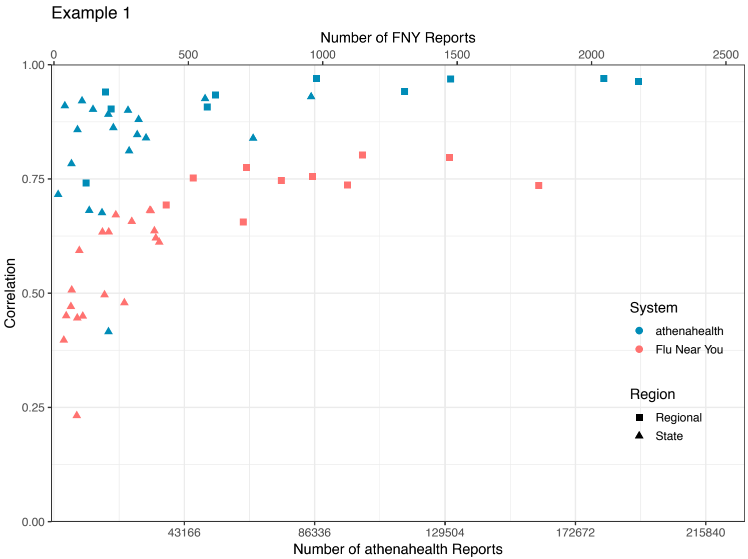
Step 12: adjust y-axis



Step 13: adjust x-axis

```
p13<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor, color="FNY"
  geom_point(aes(x=per.athena.reports, y=athena.cdc.cor, color
  theme_bw()+
  scale_color_manual(name="System",labels = c("athenahealth",
  scale_shape_manual(name="Region",labels=c("Regional", "State
  scale_y_continuous( limits = c(0,1), expand = c(0,0) )+
  ylab("Correlation")+
  xlab("Number of athenahealth Reports")+
  scale_x_continuous( limits = c(-0.1,26.5), expand = c(0,0),
          position="bottom", breaks = c(5, 10, 15, 20, 25),
          labels = c(43166, 86336, 129504, 172672, 215840),
          sec.axis= sec_axis(~.*97, name="Number of FNY Report
  ggtitle("Example 1") +
  theme(legend.justification=c(1,1),
        legend.position=c(0.98,0.5),
        legend.key = element_rect(fill="transparent"),
```

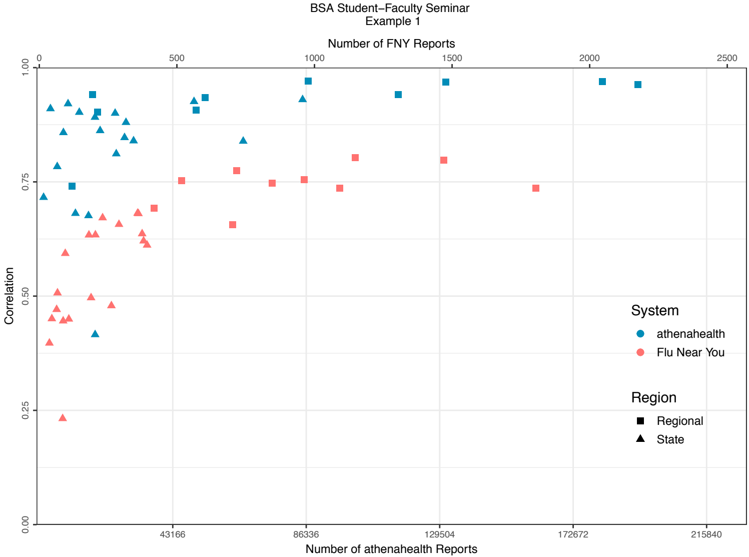
Step 13: adjust x-axis



Step 14: make adjustments to axis labels and text

```
p14<- ggplot(data1)+
  geom_point(aes(x=per.fny.reports, y=fny.cdc.cor, color="FNY
  scale_color_manual(name="System",labels = c("athenahealth",
  scale_x_continuous( limits = c(-0.1,26.5), expand = c(0,0),
  ggtitle("BSA Student-Faculty Seminar \n Example 1") +
  theme(legend.justification=c(1,1),
        legend.position=c(0.98,0.5),
        legend.key = element_rect(fill="transparent"),
        legend.background = element_rect(fill="transparent"),
        legend.key.size = unit(0.5, "cm"),
        plot.title = element_text(hjust = 0.5, size=9),
        axis.text.x=element_text(size=7),
        axis.text.y=element_text(angle=90, hjust=0.5, size=7)
        axis.title=element_text(size=9),
        text=element_text(family="sans"),
        panel.grid.minor.x = element_blank())
```

Step 14: make adjustments to axis labels and text



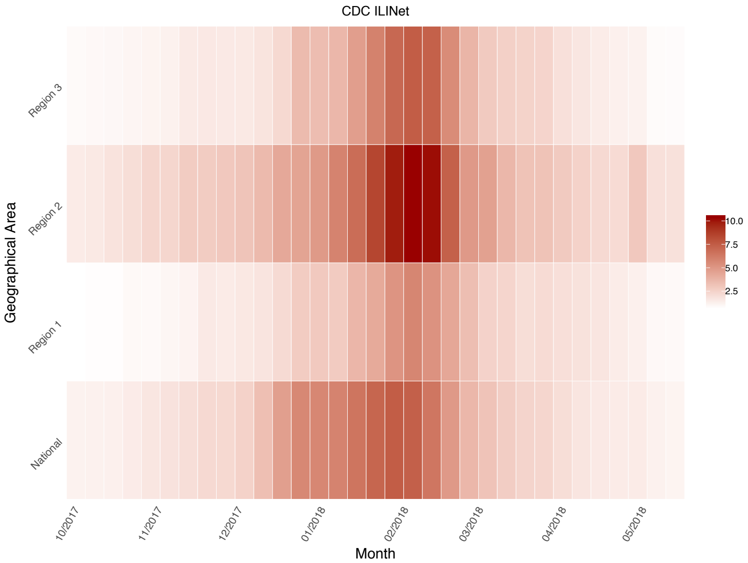
Step 15: Save the last plot created as pdf

```
ggsave("~/Documents/BU/R/ggplot2/example1.pdf",
       width = 12, height = 12, units = "cm")
```

Example 2 - Combine mutliple graphs

```
##      week_of   region   mmwr weighted_ili  per_ili   ath_il
## 1 2017-10-02 National 201740     1.265890 1.851327 0.765356
## 2 2017-10-02 Region 1 201740     0.714489 1.826484 0.625372
## 3 2017-10-02 Region 2 201740     1.529640 1.434720 0.826187
## 4 2017-10-02 Region 3 201740     0.926452 2.125850 0.818757
```

Time series heat map



Create a function

```
heat_series<-function(filler, titler){
ploty <- ggplot()+
  geom_tile(aes(y=data2$region, x=data2$week_of,fill = filler
  scale_fill_gradient(low="white", high= "#990000") +
  scale_x_date(expand = c(0,0),date_breaks = "1 month", date_
  scale_y_discrete(expand = c(0, 0)) + theme_bw() +
  ggtitle(titler)+ ylab("Geographical Area")+ xlab("Month")+
  theme(panel.grid=element_blank(), panel.border=element_blan
        legend.position="right", legend.title = element_blank
        legend.key.size = unit(0.2, "in"), legend.text = eleme
        axis.ticks = element_blank(), plot.title = element_te
        axis.text.x=element_text(angle=60, hjust=1, size=8),
        axis.text.y = element_text(angle=45, hjust = 1, size=
return(ploty)
}
```

Call the function

```r
heat1<-heat_series(data2$weighted_ili, "CDC ILINet")
heat2<-heat_series(data2$per_ili, "Flu Near You")
heat3<-heat_series(data2$ath_ili, "athenahealth")
heat4<-heat_series(data2$twe_ili, "HealthTweets.org")
```

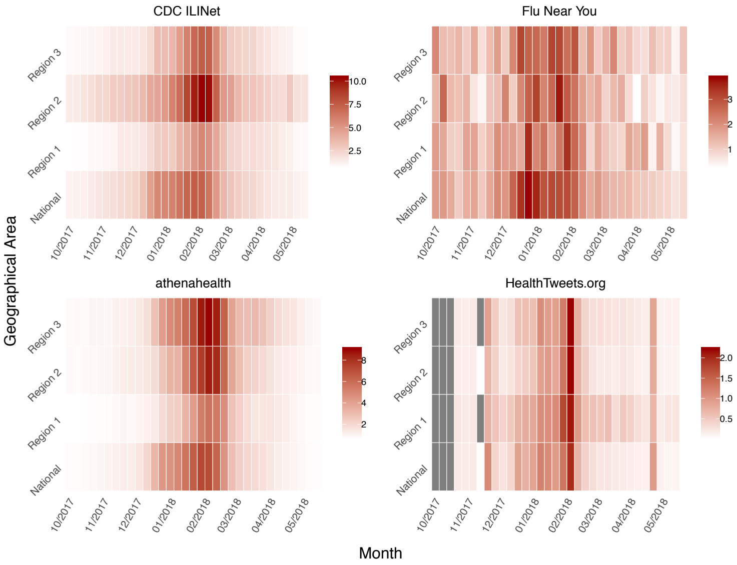
Combine into one image

```
library(gridExtra)
library(grid)

pdf(file='~/Documents/BU/R/ggplot2/example2a.pdf',width=8, he
grid.arrange(arrangeGrob(heat1 + theme(axis.title = element_b
                         heat2 + theme(axis.title = element_b
                         heat3 + theme(axis.title = element_b
                         heat4 + theme(axis.title = element_b
                         left = textGrob("Geographical Area",
                         bottom = textGrob("Month", hjust = 0
dev.off()
```

```
## pdf
##   2
```

Combine into one image



Time series heat map - categorical variable

```
# Create color buckets
data2$colorBuckets1 <- as.factor(as.numeric(
  cut(data2$weighted_ili,c(0, 1.0, 2.0, 3.0, 4.0 ,20))))
data2$colorBuckets2 <- as.factor(as.numeric(
  cut(data2$per_ili,c(0, 1.0, 2.0, 3.0, 4.0 ,20))))
data2$colorBuckets3 <- as.factor(as.numeric(
  cut(data2$ath_ili,c(0, 1.0, 2.0, 3.0, 4.0 ,20))))
data2$colorBuckets4 <- as.factor(as.numeric(
  cut(data2$twe_ili,c(0, 1.0, 2.0, 3.0, 4.0 ,20))))
```

Create a function

```
heat_series2<-function(bucket, titler){
  ploty <- ggplot()+
    geom_tile(aes(y=data2$region, x=data2$week_of,fill = bucke
    scale_fill_manual(
      values=c("#99CCFF", "#3399FF", "#0066CC","#004C99","#003
      name="Percent ILI",
      labels=c("0-1.0", "1.01-2.0", "2.01-3.0", "3.01-4.0",">4
    theme_bw() +    ggtitle(titler)+
    scale_x_date(expand = c(0,0),date_breaks = "1 month", date
    scale_y_discrete(expand = c(0, 0)) +
    ylab("Geographical Area")+ xlab("Month")+
    theme(panel.grid=element_blank(), panel.border=element_bla
          legend.position="bottom", legend.title = element_bla
          legend.key.size = unit(0.2, "in"), legend.text = ele
          axis.ticks = element_blank(),
          axis.text.x=element_text(angle=60, hjust=1, size=8)
          axis.text.y = element_text(angle=45, hjust = 1, size
```

Call the function

```
heat1a<-heat_series2(data2$colorBuckets1, "CDC ILINet")
heat2a<-heat_series2(data2$colorBuckets2, "Flu Near You")
heat3a<-heat_series2(data2$colorBuckets3, "athenahealth")
heat4a<-heat_series2(data2$colorBuckets4, "HealthTweets.org")
```

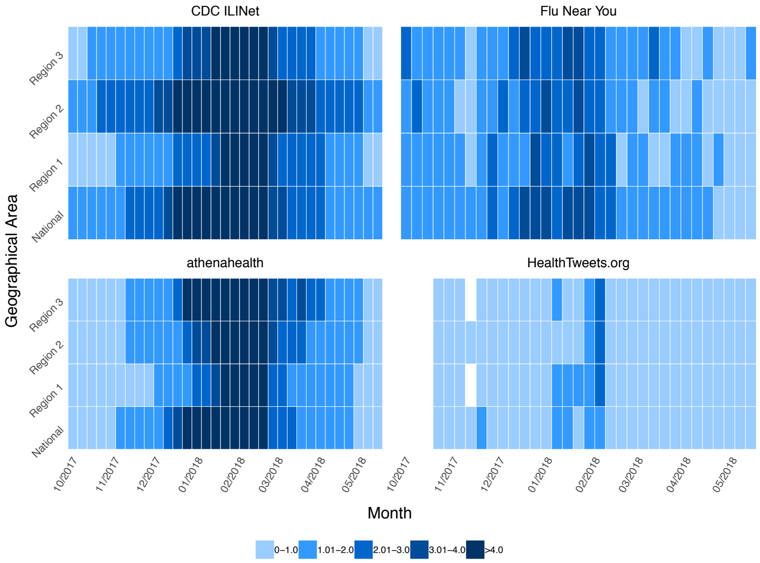
Combine into one image - use only one legend

```
library(gtable)

legend = gtable_filter(ggplotGrob(heat1a), "guide-box")

pdf(file='~/Documents/BU/R/ggplot2/example2b.pdf',width=8, he
grid.arrange(arrangeGrob(
  heat1a + theme(axis.title = element_blank(), legend.positio
                 axis.text.x = element_blank()),
  heat2a + theme(axis.title = element_blank(), legend.positio
                 axis.text = element_blank()),
  heat3a + theme(axis.title = element_blank(), legend.positio
  heat4a + theme(axis.title = element_blank(), legend.positio
                 axis.text.y = element_blank()),
  left = textGrob("Geographical Area", rot = 90, vjust = 1),
   bottom = textGrob("Month", hjust = 0.5), ncol=2),
   legend, nrow=2,heights=c(10, 1))
dev.off()
```

Combine into one image - use only one legend


CDC ILINet
Flu Near You
athenahealth
HealthTweets.org
Geographical Area
Region 3
Region 2
Region 1
National
10/2017
11/2017
12/2017
01/2018
02/2018
03/2018
04/2018
05/2018
Month
>4.0