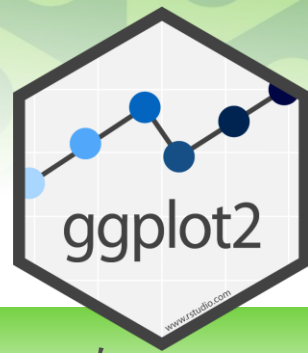


Οπτικοποιήσεις με το πακέτο ggplot2 :: Σύντομος Οδηγός

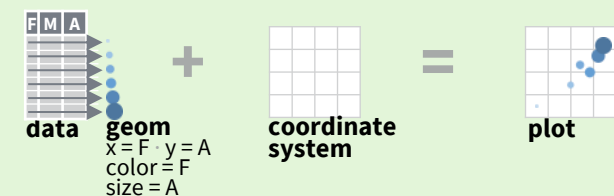


Εισαγωγή

Το πακέτο **ggplot2** βασίζεται στη γραμματική των γραφημάτων (**grammar of graphics**), στην οποία μπορεί κάποιος να δημιουργήσει γραφήματα χρησιμοποιώντας τους ίδιους παράγοντες: τα δεδομένα (**data**), ένα σύστημα συντεταγμένων (**coordinate system**) και τις γεωμετρίες (**geoms**) δηλαδή τον τρόπο που θα οπτικοποιηθούν τα δεδομένα.



Για τις τιμές του γραφήματος υπάρχουν ιδιότητες αισθητικής του γραφήματος στην γεωμετρία (**aesthetics**) όπως το μέγεθος (**size**), το χρώμα (**color**) και θέσεις των **x** και **y**.



Παρακάτω ακολουθεί έναν πρότυπο δημιουργίας ενός γραφήματος.

```
ggplot (data = <DATA>) +  
  <GEOM_FUNCTION>(mapping = aes(<MAPPINGS>),  
  stat = <STAT>, position = <POSITION>) +  
  <COORDINATE_FUNCTION> +  
  <FACET_FUNCTION> +  
  <SCALE_FUNCTION> +  
  <THEME_FUNCTION>
```

Απαιτείται
Δεν απαιτείται, δίνονται προεπιλογές

ggplot(data = mpg, aes(x = cty, y = hwy)) Με την εντολή αυτή ξεκινάμε κάθε γράφημα και συμπληρώνουμε με αυτή στρώματα γεωμετρίας (**geoms**) για να το ολοκληρώσουμε.

last_plot() Επιστρέφει το τελευταίο γράφημα.

ggsave("plot.png", width = 5, height = 5) Αποθηκεύει το τελευταίο γράφημα σε 5' x 5' αρχείο με όνομα "plot.png" στον κατάλογο εργασίας. Η επέκταση του αρχείου δίνεται μαζί με το όνομά του.

Aes

Συνήθειες αισθητικές (**aesthetics**).

Χρώμα (**color**) και γέμισμα (**fill**) - κείμενο ("red", "#RRGGBB")

Είδος γραμμής (**linetype**) - ακέραιος ή κείμενο (0 = "blank", 1 = "solid", 2 = "dashed", 3 = "dotted", 4 = "dotdash", 5 = "longdash", 6 = "twodash")

Τελείωμα γραμμής (**lineend**)- κείμενο ("round", "butt", or "square")

Ένωση γραμμών(**linejoin**)- κείμενο ("round", "mitre", or "bevel")

Μέγεθος (**size**)- ακέραιος (πάχος σε χιλιοστά)

Σχήμα (**shape**)- ακέραιος/όνομα σχήματος

ή χαρακτήρας ("a")

0 1 2 3 4 5 6 7 8 9 10 11 12
□ ○ △ + × ◇ ▽ ☆ ✱ ✳ ✴ ✵
13 14 15 16 17 18 19 20 21 22 23 24 25
⊠ ⊡ ⊢ ⊣ ⊤ ⊥ ⊦ ⊧ ⊨ ⊩ ⊪ ⊫ ⊬



Γεωμετρίες (Geoms)

Μια συνάρτηση γεωμετρίας (**geom**) χρησιμοποιείται για να καθορισθεί ο τύπος γραφήματος για τα δεδομένα και η ιδιότητα αισθητικής (**aesthetic**) των γεωμετριών (**geoms**) χρησιμοποιείται για να οριστεί ο τρόπος απεικόνισης των μεταβλητών που θα χρησιμοποιηθούν στο γράφημα. Κάθε συνάρτηση θεωρείται ως ένα στρώμα (**layer**).

Πρωτογενείς Γραφικές Παράμετροι

```
a <- ggplot(economics, aes(date, unemploy))  
b <- ggplot(seals, aes(x = long, y = lat))
```

a + geom_blank() and **a + expand_limits()**
Καθορίζει ότι τα όρια περιλαμβάνουν τιμές σε όλα τα γραφήματα.

b + geom_curve(aes(yend = lat + 1, xend = long + 1, curvature = 1)) - Παράμετροι: x, xend, y, yend, alpha, angle, color, curvature, linetype, size

a + geom_path(lineend = "butt", linejoin = "round", linemitre = 1)
Παράμετροι: x, y, alpha, color, group, linetype, size

a + geom_polygon(aes(alpha = 50)) - Παράμετροι: x, y, alpha, color, fill, group, subgroup, linetype, size

b + geom_rect(aes(xmin = long, ymin = lat, xmax = long + 1, ymax = lat + 1)) - Παράμετροι: xmax, xmin, ymax, ymin, alpha, color, fill, linetype, size

a + geom_ribbon(aes(ymin = unemploy - 900, ymax = unemploy + 900)) - Παράμετροι: x, ymax, ymin, alpha, color, fill, group, linetype, size

Ευθύγραμμα Τμήματα

Συνήθειες παράμετροι: x, y, alpha, color, linetype, size

b + geom_abline(aes(intercept = 0, slope = 1))
b + geom_hline(aes(yintercept = lat))
b + geom_vline(aes(xintercept = long))

b + geom_segment(aes(yend = lat + 1, xend = long + 1))
b + geom_spoke(aes(angle = 1:1155, radius = 1))

Μία Μεταβλητή - Συνεχής

```
c <- ggplot(mpg, aes(hwy)); c2 <- ggplot(mpg)
```

c + geom_area(stat = "bin")
Παράμετροι: x, y, alpha, color, fill, linetype, size

c + geom_density(kernel = "gaussian")
Παράμετροι: x, y, alpha, color, fill, group, linetype, size, weight

c + geom_dotplot()
Παράμετροι: x, y, alpha, color, fill

c + geom_freqpoly()
Παράμετροι: x, y, alpha, color, group, linetype, size

c + geom_histogram(binwidth = 5)
Παράμετροι: x, y, alpha, color, fill, linetype, size, weight

c2 + geom_qq(aes(sample = hwy))
Παράμετροι: x, y, alpha, color, fill, linetype, size, weight

Διακριτή

```
d <- ggplot(mpg, aes(fl))
```

d + geom_bar()
Παράμετροι: x, alpha, color, fill, linetype, size, weight

Δύο Μεταβλητές

Συνεχείς και οι δυο

```
e <- ggplot(mpg, aes(cty, hwy))
```

e + geom_label(aes(label = cty), nudge_x = 1, nudge_y = 1) - Παράμετροι: x, y, label, alpha, angle, color, family, fontface, hjust, lineheight, size, vjust

e + geom_point()
Παράμετροι: x, y, alpha, color, fill, shape, size, stroke

e + geom_quantile()
Παράμετροι: x, y, alpha, color, group, linetype, size, weight

e + geom_rug(sides = "bl")
Παράμετροι: x, y, alpha, color, linetype, size

e + geom_smooth(method = lm)
Παράμετροι: x, y, alpha, color, fill, group, linetype, size, weight

e + geom_text(aes(label = cty), nudge_x = 1, nudge_y = 1) - Παράμετροι: x, y, label, alpha, angle, color, family, fontface, hjust, lineheight, size, vjust

Μία διακριτή, μία συνεχής

```
f <- ggplot(mpg, aes(class, hwy))
```

f + geom_col()
Παράμετροι: x, y, alpha, color, fill, group, linetype, size

f + geom_boxplot()
Παράμετροι: x, y, lower, middle, upper, ymax, ymin, alpha, color, fill, group, linetype, shape, size, weight

f + geom_dotplot(binaxis = "y", stackdir = "center") - Παράμετροι: x, y, alpha, color, fill, group

f + geom_violin(scale = "area")
Παράμετροι: x, y, alpha, color, fill, group, linetype, size, weight

Δύο Διακριτές

```
g <- ggplot(diamonds, aes(cut, color))
```

g + geom_count()
Παράμετροι: x, y, alpha, color, fill, shape, size, stroke

e + geom_jitter(height = 2, width = 2)
Παράμετροι: x, y, alpha, color, fill, shape, size

Τρεις Μεταβλητές

```
seals$z <- with(seals, sqrt(delta_long^2 + delta_lat^2)); l <- ggplot(seals, aes(long, lat))
```

l + geom_contour(aes(z = z))
Παράμετροι: x, y, z, alpha, color, group, linetype, size, weight

l + geom_contour_filled(aes(fill = z))
Παράμετροι: x, y, alpha, color, fill, group, linetype, size, subgroup

Συνεχής Διμεταβλητή Κατανομή

```
h <- ggplot(diamonds, aes(carat, price))
```

h + geom_bin2d(binwidth = c(0.25, 500))
Παράμετροι: x, y, alpha, color, fill, linetype, size, weight

h + geom_density_2d()
Παράμετροι: x, y, alpha, color, group, linetype, size

h + geom_hex()
Παράμετροι: x, y, alpha, color, fill, size

Συνεχής Συνάρτηση

```
i <- ggplot(economics, aes(date, unemploy))
```

i + geom_area()
Παράμετροι: x, y, alpha, color, fill, linetype, size

i + geom_line()
Παράμετροι: x, y, alpha, color, group, linetype, size

i + geom_step(direction = "hv")
Παράμετροι: x, y, alpha, color, group, linetype, size

Οπτικοποιώντας το σφάλμα

```
df <- data.frame(grp = c("A", "B"), fit = 4:5, se = 1:2)  
j <- ggplot(df, aes(grp, fit, ymin = fit - se, ymax = fit + se))
```

j + geom_crossbar(fatten = 2) - Παράμετροι: x, y, ymax, ymin, alpha, color, fill, group, linetype, size

j + geom_errorbar() - Παράμετροι: x, ymax, ymin, alpha, color, group, linetype, size, width
Also **geom_errorbarh()**.

j + geom_linerange()
Παράμετροι: x, ymin, ymax, alpha, color, group, linetype, size

j + geom_pointrange() - Παράμετροι: x, y, ymin, ymax, alpha, color, fill, group, linetype, shape, size

Χάρτες

```
data <- data.frame(murder = USArrests$Murder,  
  state = tolower(rownames(USArrests)))  
map <- map_data("state")  
k <- ggplot(data, aes(fill = murder))
```

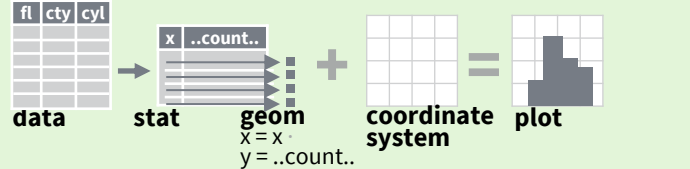
k + geom_map(aes(map_id = state), map = map) + expand_limits(x = map\$long, y = map\$lat)
map_id, alpha, color, fill, linetype, size

l + geom_raster(aes(fill = z), hjust = 0.5, vjust = 0.5, interpolate = FALSE)
Παράμετροι: x, y, alpha, fill

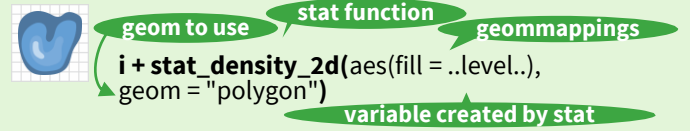
l + geom_tile(aes(fill = z))
Παράμετροι: x, y, alpha, color, fill, linetype, size, width

Στατιστική(Stats)

Ένας άλλος τρόπος για την προσθήκη νέου στρώματος (layer). Μια στατιστική (stat) δημιουργεί νέες μεταβλητές για προβολή (π.χ., πλήθος-count, αναλογίες-prop).



Οπτικοποιείτε μια στατιστική αλλάζοντας την προεπιλογή stat της γεωμετρίας, π.χ. **geom_bar(stat="count")** ή χρησιμοποιώντας μία συνάρτηση στατιστικής, π.χ. **stat_count(geom="bar")**, η οποία θα καλεί την προεπιλογή της αντίστοιχης γεωμετρίας για τον σχεδιασμό του στρώματος (ισοδυναμεί με μια συνάρτηση γεωμετρίας). Χρησιμοποιείτε την παράμετρο **..name..** για να αντιστοιχίσετε την κατάλληλη γεωμετρία.



c + stat_bin(binwidth = 1, boundary = 10)
x, y | ..count.., ..ncount.., ..density.., ..ndensity..
c + stat_count(width = 1) **x, y** | ..count.., ..prop..

c + stat_density(adjust = 1, kernel = "gaussian")
x, y | ..count.., ..density.., ..scaled..
e + stat_bin_2d(bins = 30, drop = T)
x, y, fill | ..count.., ..density..
e + stat_bin_hex(bins = 30) **x, y, fill** | ..count.., ..density..

e + stat_density_2d(contour = TRUE, n = 100)
x, y, color, size | ..level..

e + stat_ellipse(level = 0.95, segments = 51, type = "t")
l + stat_contour(aes(z = z)) **x, y, z, order** | ..level..
l + stat_summary_hex(aes(z = z), bins = 30, fun = max)
x, y, z, fill | ..value..

l + stat_summary_2d(aes(z = z), bins = 30, fun = mean)
x, y, z, fill | ..value..
f + stat_boxplot(coef = 1.5)
x, y | ..lower.., ..middle.., ..upper.., ..width.., ..ymin.., ..ymax..

f + stat_ydensity(kernel = "gaussian", scale = "area") **x, y** | ..density.., ..scaled.., ..count.., ..n.., ..violinwidth.., ..width..
e + stat_ecdf(n = 40) **x, y** | ..x.., ..y..

e + stat_quantile(quantiles = c(0.1, 0.9), formula = y ~ log(x), method = "rq") **x, y** | ..quantile..

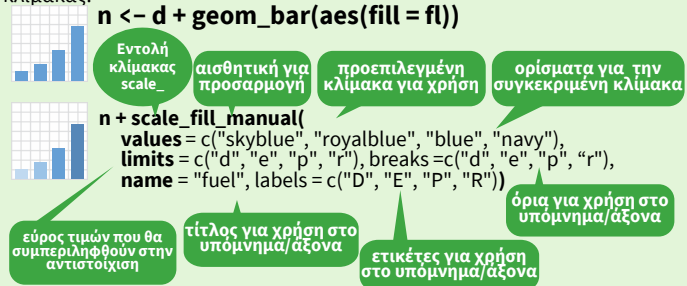
e + stat_smooth(method = "lm", formula = y ~ x, se = T, level = 0.95) **x, y** | ..se.., ..x.., ..y.., ..ymin.., ..ymax..
ggplot() + **xlim**(-5, 5) + **stat_function**(fun = dnorm, n = 20, geom = "point") **x** | ..x.., ..y..

ggplot() + **stat_qq**(aes(sample = 1:100))
x, y, sample | ..sample.., ..theoretical..

e + stat_sum() **x, y, size** | ..n.., ..prop..
e + stat_summary(fun.data = "mean_cl_boot")
h + stat_summary_bin(fun = "mean", geom = "bar")
e + stat_identity()
e + stat_unique()

Κλίμακες(Scales)

Μπορεί να γίνει αντικατάσταση προεπιλογών με το πακέτο **scales**. Οι **κλίμακες (Scales)** αντιστοιχούν τις τιμές των δεδομένων με τις παραμέτρους οπτικοποίησης μιας αισθητικής (aesthetic). Για να αλλάξετε την προεπιλεγμένη αντιστοίχιση μπορείτε να προσθέσετε μια νέα εντολή κλίμακας.



Κλίμακες Γενικής Χρήσης

scale_*_continuous() – Αντιστοίχιση συνεχών μεταβλητών σε συνεχείς μεταβλητές οπτικοποίησης.
scale_*_discrete() – Αντιστοίχιση διακριτών μεταβλητών σε διακριτές μεταβλητές οπτικοποίησης.
scale_*_binned() – Αντιστοιχίστε συνεχείς τιμές σε διακριτές κλάσεις διαστημάτων.
scale_*_identity() – Χρήση δεδομένων ως οπτικές παράμετροι.
scale_*_manual(values = c()) – Αντιστοίχιση διακριτών τιμών από συγκεκριμένες παραμέτρους οπτικοποίησης.
scale_*_date(date_labels = "%m/%d"), date_breaks = "2 weeks") – Αντιμετώπιση τιμών δεδομένων ως ημερομηνίες.
scale_*_datetime() – Αντιμετώπιση τιμών δεδομένων ως ημερομηνία και ώρα.
Το ίδιο όπως το **scale_*_date()**. Δείτε ?strptime για περισσότερες επιλογές ετικέτας (label formats).

X & Y Κλίμακες Θέσης (LOCATION SCALES)

Χρησιμοποιείτε για x ή y αισθητική (π.χ. για x) **scale_x_log10()** – Σχεδίαση άξονα x στην κλίμακα log10.
scale_x_reverse() – Αντιστρέψτε την κατεύθυνση του άξονα x.
scale_x_sqrt() – Σχεδίαση άξονα x στην κλίμακα της τετραγωνικής ρίζας.

Κλίμακες Χρωμάτων και Γεμίσματος (Διακριτές)

n + scale_fill_brewer(palette = "Blues")
Για περισσότερα χρώματα παλέτας: RColorBrewer::display.brewer.all()
n + scale_fill_grey(start = 0.2, end = 0.8, na.value = "red")

Κλίμακες Χρωμάτων και Γεμίσματος (Συνεχείς)

o <- c + geom_dotplot(aes(fill = ..x..))
o + scale_fill_distiller(palette = "Blues")
o + scale_fill_gradient(low="red", high="yellow")
o + scale_fill_gradient2(low = "red", high = "blue", mid = "white", midpoint = 25)
o + scale_fill_gradientn(colors = topo.colors(6))
Επίσης: rainbow(), heat.colors(), terrain.colors(), cm.colors(), RColorBrewer::brewer.pal()

Κλίμακες για Σχήμα και Μέγεθος

p <- e + geom_point(aes(shape = fl, size = cyl))
p + scale_shape() + **scale_size()**
p + scale_shape_manual(values = c(3:7))
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
□○△+×◇▽※✱✱✱✱✱✱✱✱✱✱□○△◇○○●●◆◆▽▽
p + scale_radius(range = c(1,6))
p + scale_size_area(max_size = 6)

Συστήματα Συντεταγμένων

r <- d + geom_bar()
r + coord_cartesian(xlim = c(0, 5)) - xlim, ylim
Το προεπιλεγμένο καρτεσιανό σύστημα συντεταγμένων.
r + coord_fixed(ratio = 1/2)
ratio, xlim, ylim - Καρτεσιανές συντεταγμένες με σταθερό λόγο διαστάσεων μεταξύ των x και y.
ggplot(mpg, aes(y = fl)) + geom_bar()
Αντιστροφή καρτεσιανών συντεταγμένων x,y.
r + coord_polar(theta = "x", direction=1)
theta, start, direction - Πολικές συντεταγμένες.
r + coord_trans(y = "sqrt") - x, y, xlim, ylim
Μετασχηματισμένες καρτεσιανές συντεταγμένες.
π + coord_quickmap()
π + coord_map(projection = "ortho", orientation = c(41, -74, 0)) - projection, xlim, ylim
Προβολές χάρτη από το πακέτο mapproj (mercator (default), azequalarea, lagrange, etc.).

Προσαρμογή Θέσης

Η προσαρμογή θέσης καθορίζει πως θα κατανεμηθούν τα επιμέρους κομμάτια των γραφημάτων.

s <- ggplot(mpg, aes(fl, fill = drv))
s + geom_bar(position = "dodge")
Τοποθετήστε τα στοιχεία δίπλα-δίπλα.
s + geom_bar(position = "fill")
Τοποθετήστε τα στοιχεία το ένα πάνω στο άλλο, κανονικοποιείτε το ύψος.
e + geom_point(position = "jitter")
Προσθήκη τυχαίου θορύβου στις θέσεις των τιμών των X και Y για την αποφυγή αλληλοεπικάλυψης.
e + geom_label(position = "nudge")
Τοποθετήστε τις ετικέτες μακριά από σημεία.
s + geom_bar(position = "stack")
Τοποθετήστε τα στοιχεία το ένα πάνω στο άλλο.

Κάθε προσαρμογή θέσης μπορεί να ενσωματωθεί ως συνάρτηση με συγκεκριμένα ορίσματα τιμών **ύψους ή/και πλάτους** ως εξής:
s + geom_bar(position = position_dodge(width = 1))

Θέματα

r + theme_bw()
Λευκό φόντο με γραμμές πλέγματος.
r + theme_classic()
r + theme_light()
r + theme_gray()
Γκρι φόντο (προεπιλεγμένο θέμα).
r + theme_linedraw()
Μίνιμαλ θέμα.
r + theme_dark()
Σκούρο για αντίθεση.
r + theme_void()
Κενό θέμα.

r + theme() Προσαρμόστε παραμέτρους του θέματος, όπως π.χ. για τον άξονα, το υπόμνημα και τα πάνελ
r + ggtitle("Title") + **theme**(plot.title.position = "plot")
r + theme(panel.background = element_rect(fill = "blue"))

Faceting

Το πάνελ γραφημάτων αποτελείται από επιμέρους γραφήματα με βάση τις τιμές μιας ή περισσότερων διακριτών μεταβλητών.

t <- ggplot(mpg, aes(cty, hwy)) + **geom_point**()
t + facet_grid(cols = **vars**(fl))
Πάνελ στηλών σύμφωνα με το fl.
t + facet_grid(rows = **vars**(year))
Πάνελ γραμμών σύμφωνα με την year.
t + facet_grid(rows = **vars**(year), cols = **vars**(fl))
Πάνελ γραμμών και στηλών.
t + facet_wrap(**vars**(fl))
Ορθογώνια πάνελ.
Η παράμετρος scales προσαρμόζει τα όρια των αξόνων στα επιμέρους πάνελ.
t + facet_grid(rows = **vars**(drv), cols = **vars**(fl), scales = "free")
Αντίστοιχα για τα όρια
"free_x" - μόνο του άξονα x
"free_y" - μόνο του άξονα y

Η παράμετρος **labeller** καθορίζει τις ετικέτες των πάνελ:

t + facet_grid(cols = **vars**(fl), **labeller** = **label_both**)

fl: c	fl: d	fl: e	fl: p	fl: r
α^c	α^d	α^e	α^p	α^r

t + facet_grid(rows = **vars**(fl), **labeller** = **label_bquote**(alpha ^ .(fl)))

Ετικέτες και Υπόμνημα

Χρησιμοποιείτε την εντολή **labs()** για να προσθέσετε ετικέτες στο γράφημά σας.
t + labs(x = "Νέα ετικέτα άξονα x", y = "Νέα ετικέτα άξονα y", title = "Προσθήκη τίτλου πάνω από το γράφημα", subtitle = "Προσθήκη υπότιτλου κάτω από τον τίτλο", caption = "Προσθήκη λεζάντας κάτω από το γράφημα", alt = "Προσθέστε εναλλακτικό κείμενο στο γράφημα",
<AES> = "Νέο <AES> τίτλο υπομνήματος")

t + annotate(geom = "text", x = 8, y = 9, label = "A")
Τοποθέτηση μιας γεωμετρίας με επιλεγμένη αισθητική (θέση και ετικέτα).

p + guides(x = guide_axis(n.dodge = 2)) **Αποφυγή αλληλεπικαλυπτόμενων υπονημάτων** guide_axis(n.dodge or angle).

n + guides(fill = "none") **Ορισμός υπομνήματος για κάθε τύπο αισθητικής:** colorbar, legend, or none (no legend).

n + theme(legend.position = "bottom")
Τοποθέτηση υπομνήματος κάτω ("bottom"), πάνω("top"), αριστερά ("left"), ή δεξιά ("right").

n + scale_fill_discrete(name = "Title", labels = c("A", "B", "C", "D", "E"))
Ορισμός τίτλου και λεζάντας υπομνήματος με μια συνάρτηση scale.

Μεγέθυνση

Χωρίς απόκομμα (προτιμάται):
t + coord_cartesian(xlim = c(0, 100), ylim = c(10, 20))
Με αποκοπή (αφαιρεί τα μη ορατά σημεία δεδομένων):
t + xlim(0, 100) + **ylim**(10, 20)
t + scale_x_continuous(limits = c(0, 100)) + **scale_y_continuous**(limits = c(0, 100))

