

Tableau de Fonctions - Cours 4 : Visualisation de Données

Nolan

November 8, 2024

Fonction	Usage	Résultat/Exemple
<code>ggplot()</code>	Initialisation de la visualisation. Utilisé pour définir les données et les mappings esthétiques globaux.	<code>ggplot(mpg, aes(x = displ, y = hwy)) + geom_point()</code>
<code>geom_point()</code>	Crée un nuage de points.	<code>ggplot(mpg, aes(x = displ, y = hwy)) + geom_point()</code>
<code>geom_line()</code>	Crée un graphique en ligne. Utile pour les séries temporelles.	<code>ggplot(economics, aes(x = date, y = unemploy)) + geom_line(color = "blue")</code>
<code>geom_bar()</code>	Crée un diagramme en barres.	<code>ggplot(mpg, aes(x = class)) + geom_bar(fill = "coral")</code>
<code>geom_histogram()</code>	Crée un histogramme.	<code>ggplot(mpg, aes(x = hwy)) + geom_histogram(binwidth = 2)</code>
<code>geom_boxplot()</code>	Crée un diagramme en boîte (box plot).	<code>ggplot(mpg, aes(x = class, y = hwy)) + geom_boxplot()</code>
<code>geom_density()</code>	Crée une courbe de densité.	<code>ggplot(penguins, aes(x = body_mass_g, fill = species)) + geom_density(alpha = 0.5)</code>
<code>geom_violin()</code>	Crée un diagramme en forme de violon pour visualiser la distribution des données.	<code>ggplot(penguins, aes(x = species, y = bill_length_mm, fill = species)) + geom_violin()</code>
<code>geom_smooth()</code>	Ajoute une ligne de tendance ou de régression.	<code>ggplot(mpg, aes(x = displ, y = hwy)) + geom_point() + geom_smooth(method = "lm", se = FALSE)</code>
<code>facet_wrap()</code>	Divise le graphique en sous-graphiques en fonction d'une variable catégorielle.	<code>ggplot(mpg, aes(x = displ, y = hwy)) + geom_point() + facet_wrap(~class)</code>
<code>scale_x_continuous()</code>	Personnalise l'échelle de l'axe X pour des valeurs continues.	<code>scale_x_continuous(name = "Poids (lbs)", breaks = seq(1, 5, by = 1))</code>
<code>scale_y_continuous()</code>	Personnalise l'échelle de l'axe Y pour des valeurs continues.	<code>scale_y_continuous(name = "Miles per gallon", trans = "log10")</code>
<code>scale_fill_continuous()</code>	Définit une échelle de couleur pour les valeurs continues, appliquée au remplissage des formes.	<code>scale_fill_continuous(low = "blue", high = "red")</code>

Fonction	Usage	Résultat/Exemple
<code>scale_fill_discrete()</code>	Définit une échelle de couleur pour les valeurs discrètes, appliquée au remplissage des formes.	<code>scale_fill_discrete(name = "Type de transmission")</code>
<code>scale_size_continuous()</code>	Utilise une variable continue pour la taille des points.	<code>ggplot(mtcars, aes(x = wt, y = mpg, size = hp)) + scale_size_continuous(name = "Puissance (ch)")</code>
<code>theme()</code>	Personnalise l'apparence globale du graphique.	<code>theme(plot.title = element_text(face = "bold", size = 16))</code>
<code>theme_minimal()</code>	Applique un thème minimaliste au graphique.	<code>ggplot(mpg, aes(x = displ, y = hwy, color = class)) + theme_minimal()</code>
<code>theme_void()</code>	Applique un thème vide (sans axes ni arrière-plan).	<code>ggplot(world, aes(x = long, y = lat, group = group)) + theme_void()</code>
<code>coord_flip()</code>	Inverse les axes X et Y.	<code>ggplot(mpg, aes(x = class, y = hwy)) + geom_boxplot() + coord_flip()</code>
<code>ggsave()</code>	Sauvegarde le graphique dans un fichier.	<code>ggsave("graphique.png", plot = p, width = 10, height = 6, dpi = 300)</code>
<code>annotate()</code>	Ajoute des annotations au graphique.	<code>annotate("text", x = 5, y = 40, label = "Tendance générale", angle = 15)</code>
<code>scale_color_viridis_c()</code>	Applique une palette de couleurs viridis pour une échelle continue.	<code>ggplot(mpg, aes(x = displ, y = hwy, color = cyl)) + scale_color_viridis_c()</code>
<code>geom_tile()</code>	Crée un graphique en tuiles, souvent utilisé pour les heatmaps.	<code>ggplot(df, aes(x = species, y = island, fill = avg.mass)) + geom_tile()</code>
<code>geom_count()</code>	Utilisé pour les données entières, montre le nombre de points superposés.	<code>ggplot(mtcars, aes(x = cyl, y = gear)) + geom_count(color = "blue")</code>
<code>geom_jitter()</code>	Ajoute un léger décalage aléatoire pour éviter le chevauchement des points.	<code>ggplot(mpg, aes(x = cyl, y = hwy)) + geom_jitter(width = 0.3)</code>
<code>geom_polygon()</code>	Utilisé pour dessiner des polygones, souvent pour les cartes.	<code>ggplot(world, aes(x = long, y = lat, group = group)) + geom_polygon(fill = "lightblue", color = "white")</code>