`stat_smooth` in `ggplot2`: A Comprehensive Guide

Introduction

The `stat_smooth` function in `ggplot2` adds smoothed conditional means or trend lines to a plot. It is commonly used to visualize relationships between variables, model predictions, or highlight overall trends in the data.

Theoretical Overview of `stat_smooth`

The `stat_smooth` function is a statistical transformation that computes a smooth line based on the data and a chosen method. The most common methods include:

- **Linear Models (LM):** Fits a straight line to the data.
- **Generalized Additive Models (GAM):** Fits a flexible curve to the data.
- **Loess (Local Regression):** Fits a non-parametric local regression.
- **Other Methods:** Allows custom functions or pre-specified models.

Key Features: 1. Automatically calculates confidence intervals around the smoothed line. 2. Flexible customization through parameters such as `method`, `span`, and `formula`. 3. Works seamlessly with aesthetic mappings to distinguish groups.

Parameters of `stat_smooth`

`stat_smooth` includes several parameters for customization:

- **method:** The smoothing method (e.g., `"lm"`, `"loess"`, `"gam"`).
- **formula:** The formula for the smoothing line (default is `y x`).
- **se:** Whether to display confidence intervals (default is `TRUE`).
- **level:** The confidence level for the interval (default is `0.95`).

- **span:** The degree of smoothing for `"loess"` (default is `0.75`).
- **fullrange:** Whether to extend the line to cover the full range of the x-axis (default is `FALSE`).
- **geom:** The geometry used to display the smooth (default is `"smooth"`).
- **position:** Adjusts the position of the smoothed line.

Examples of `stat_smooth` Usage

Example 1: Linear Fit (Default)

The default smoothing method is linear regression (`"lm"`).

```
library(ggplot2)

ggplot(data = mtcars, aes(x = wt, y = mpg)) +
  geom_point() +
  stat_smooth(method = "lm", color = "red", se = TRUE) +
  labs(title = "Linear Fit with Confidence Interval", x = "
      Weight", y = "Miles per Gallon") +
  theme_minimal()
```

Example 2: Loess Smoothing

Using local regression (`"loess"`) to fit a smooth curve.

```
ggplot(data = mtcars, aes(x = wt, y = mpg)) +
  geom_point() +
  stat_smooth(method = "loess", color = "blue", se = TRUE,
      span = 0.5) +
  labs(title = "Loess Smoothing with Span = 0.5", x = "
      Weight", y = "Miles per Gallon") +
  theme_light()
```

Example 3: Removing Confidence Intervals

Turning off the display of confidence intervals.

```
ggplot(data = mtcars, aes(x = wt, y = mpg)) +
  geom_point() +
  stat_smooth(method = "lm", se = FALSE, color = "green") +
  labs(title = "Linear Fit without Confidence Interval", x =
       "Weight", y = "Miles per Gallon") +
  theme_classic()
```

Example 4: Using GAM for Flexible Curves

Using "gam" for a generalized additive model.

```
library(mgcv)

ggplot(data = mtcars, aes(x = wt, y = mpg)) +
  geom_point() +
  stat_smooth(method = "gam", formula = y ~ s(x), color = "
      purple", se = TRUE) +
  labs(title = "Generalized Additive Model (GAM)", x = "
      Weight", y = "Miles per Gallon") +
  theme_minimal()
```

Example 5: Grouping with Aesthetics

Adding separate smoothed lines for groups.

```
ggplot(data = mtcars, aes(x = wt, y = mpg, color = factor(
    cyl))) +
  geom_point() +
  stat_smooth(method = "lm", se = FALSE) +
  labs(title = "Grouped Linear Fit by Cylinders", x = "
      Weight", y = "Miles per Gallon") +
  theme_light()
```

Example 6: Extending the Range

Extending the smoothing line to the full range of the x-axis.

```
ggplot(data = mtcars, aes(x = wt, y = mpg)) +
  geom_point() +
  stat_smooth(method = "lm", fullrange = TRUE, color = "red"
      , se = FALSE) +
  labs(title = "Extended Linear Fit", x = "Weight", y = "
      Miles per Gallon") +
  theme_classic()
```

Example 7: Customizing Confidence Levels

Changing the confidence level for the interval.

```
ggplot(data = mtcars, aes(x = wt, y = mpg)) +
  geom_point() +
  stat_smooth(method = "lm", level = 0.99, color = "orange")
      +
  labs(title = "Linear Fit with 99% Confidence Interval", x
      = "Weight", y = "Miles per Gallon") +
  theme_minimal()
```

Tips for Using `stat_smooth`

- Use `method = "loess"` for smaller datasets or data with non-linear patterns.
- Use `method = "lm"` for large datasets or when fitting a straight line.
- Combine `stat_smooth` with grouping aesthetics (`color`, `linetype`) for comparisons.
- Adjust `span` in `"loess"` for finer or coarser smoothing.
- Preprocess data with `dplyr` for advanced modeling or subsetting before applying `stat_smooth`.