Exponential smoothing

Exponential smoothing applies a weighted average where weights decrease exponentially for past observations, putting more weights on recent values to model trends.

For a time series X_t, the exponentially smoothed value S_t is defined as:

$$S_t = \alpha X_t + (1 - \alpha) S_{t-1}, \quad \alpha \in [0, 1]$$

where α is the smoothing parameter, and S_0 is an initial value (often X_1).

Moving Average smoothing

Moving Average smoothing is a smoothing technique that averages a fixed number of consecutive observations to reduce random fluctuations and outputs a smoothed series.

For a time series X_t and window size m, the smoothed value S_t is:

$$S_t = \frac{1}{m} \sum_{j=0}^{m-1} X_{t-j},$$

for $t \geq m$, where the window size m determines the level of smoothing.

Kernel smoothing

Kernel smoothing is a smoothing technique using a weighted average, with weights determined by a Gaussian kernel to give higher importance to nearby observations.

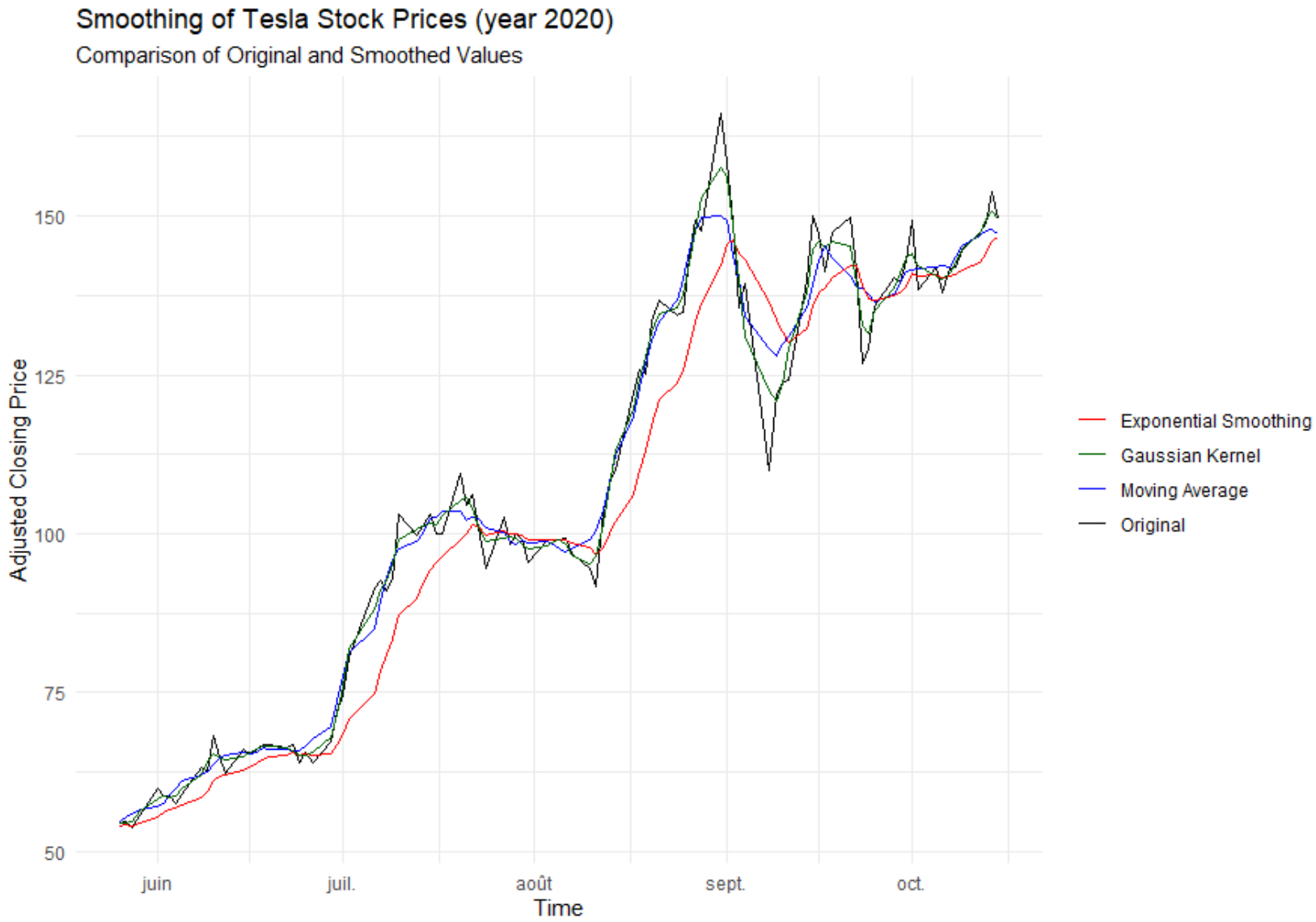
For a time series X_t, the smoothed value S_t is:

$$S_t = \sum_s K_h(t-s)X_s,$$

where K_h is a Gaussian Kernel with bandwidth h, controlling the smoothness and defined as:

$$K_h(t-s) = \frac{1}{\sqrt{2\pi h^2}} exp\left(-\frac{(t-s)^2}{2h^2} \right)$$

Smoothing of Tesla Stock prices



Interpretations

Moving Average applies uniform weights within a fixed window, Exponential Smoothing gives exponentially decreasing weights, and Gaussian Kernel uses weights from a bell-shaped Gaussian function for localized smoothing.

Exponential Smoothing adapts better to trends and dynamic changes, whereas Moving Average is rigid, and Gaussian Kernel is ideal for handling irregular or noisy data with localized smoothing.

Moving Average and Gaussian Kernel truncate results at edges due to windowing, while Exponential Smoothing does not require truncation as it uses recursive calculations.

Moving Average and Exponential Smoothing are computationally efficient, while Gaussian Kernel requires more computation and careful bandwidth selection for optimal performance.