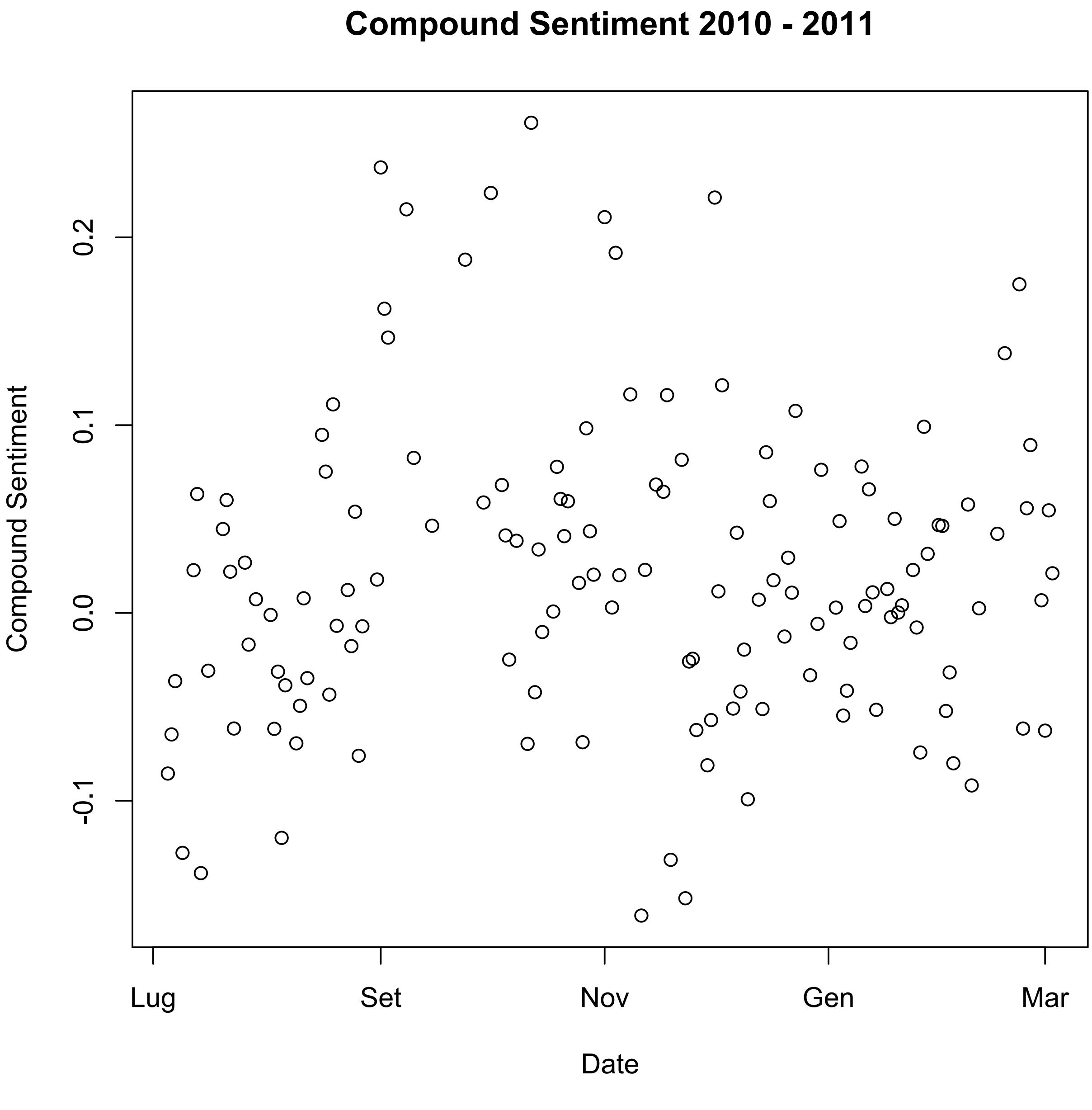
Removing Sentiment noise using the Kalman Filter

Sentiment is noisy.
This peculiar sentiment characteristic is shown in the plot below, where a 9 months sample of daily VADER extracted compound sentiment scores is plotted.



Given sentiment noisy nature, is therefore mandatory to remove noise from sentiment observations as much as possible.
Sentiment scores can in fact be considered as noisy observations of the actual but unobserved news sentiment, therefore sentiment scores need to be filtered in order to estimate the unobserved sentiment state.

The Kalman Filter

The Kalman filter is used to estimate the process (also called state) underlying a set of measurements.
Consider the following State-Space Model

$$\vec{x}_{i+1} = A\vec{x}_i + Gu_i$$

$$\vec{y}_i = H_i\vec{x}_i$$

In this model only $\vec{y}_i$ is directly observable, while $\vec{x}_i$ is unobservable: this situation leads to the Observer Design Problem, namely the basic problem of estimating an internal state of a linear system by observing just the system outputs.

Consider now that we want to estimate the following state starting from a set of measurements. The model is

$$1.\quad x_k = Ax_{k-1} + Bu_k + w_{k-1}$$

$$2.\quad z_k = Hx_k + v_k$$

Equation 1 describes the process, while equation 2 describes the measurements. Moreover we say that w_k, v_k are respectively process and measurement noises: they are assumed to be *independent*, *white* and *normally distributed*. We can define w_k, v_k distributions as

$$p(w) \sim N(0, Q)$$

$$p(v) \sim N(0, R)$$

where Q, R are respectively *process noise* and *measurement noise* error covariances.
Moreover it has to be underlined that matrices A, H, as the covariances Q, R, may change at each step, but in the discrete case they are assumed to be constant.

In the discrete case the goal of the Kalman Filter is to find the K that minimizes the *a posteriori* estimate error covariance $P_k = E\left(e_k e_k^{|}\right)$, where e_k is the prediction error $e_k = x_k - \hat{x}_k$. $\hat{x}_k$ is the *a posteriori* prediction $\hat{x}_k = \hat{x}_k^- + K(z_k - H\hat{x}_k^-)$ based on *a priori* estimates $\hat{x}_k^-$. In other words Kalman Filter minimises all estimated parameters mean square error, if Gaussian error is provided.
In the discrete case K is found by

$$\min e_k = \min(\,x_k - \hat{x}_k)$$

by minimizing the prediction error

$$K_k = P_k^- H^{|}(HP_k^- H^{|} + R)^{-1}$$

which is used in the *a posteriori* state prediction $\hat{x}_k$, that is

$$\hat{x}_k = \hat{x}_k^- + K(z_k - H\hat{x}_k^-)$$

In particular as R goes to 0, Kalman filter weights the residual $(z_k - H\hat{x}_k^-)$ more heavily, while as the *a priori* estimate error covariance P_k^- goes to 0, Kalman filter weights the residual less heavily. In other words

$$\lim_{R_k \to 0} K_k = H^{-1}$$

$$\lim_{P_k \to 0} K_k = 0$$

Kalman Filter Algorithm:

Kalman Filter algorithm is a two-step procedure where a *Time Update* is followed by a *Measurement Update*. *Time Update* projects forward in time current state and error covariance estimates in order to obtain the *a priori* estimates for the next time step. In *Measurement Update* a new observation is incorporated into the *a priori* estimate to obtain in improved *a posteriori* estimate. The algorithm is therefore a *predictor-corrector* one.

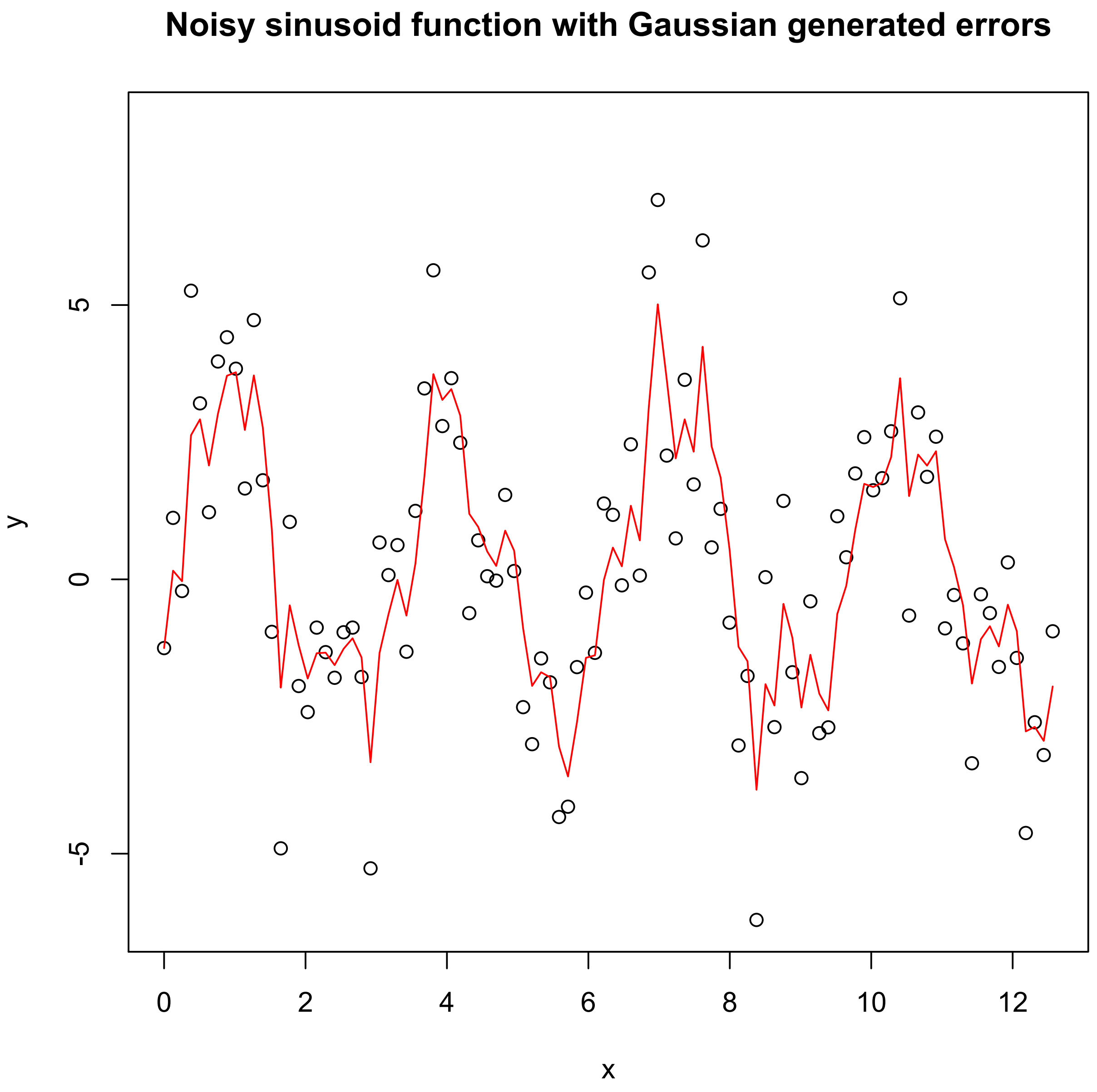
The algorithm starts after initial estimates $\hat{x}_{k-1} = x_0$ and $P_{k-1} = P_0$ are given for the first Time Update, then it follows the below described cycle

Time Update	***Measurement Update***
1) Project the state $\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k$ 2) Project the error covariance ahead $P_k^- = AP_{k-1}A^{\mid} + Q$	1) Compute Kalman K $K_k = P_k^- H^{\mid}(HP_k^- H^{\mid} + R)^{-1}$ 2) Update the estimate with observations z_K $\hat{x}_k = \hat{x}_k^- + K(z_k - H\hat{x}_k^-)$ 3) Update the error covariance $P_k = (I - K_k H)P_k^-$

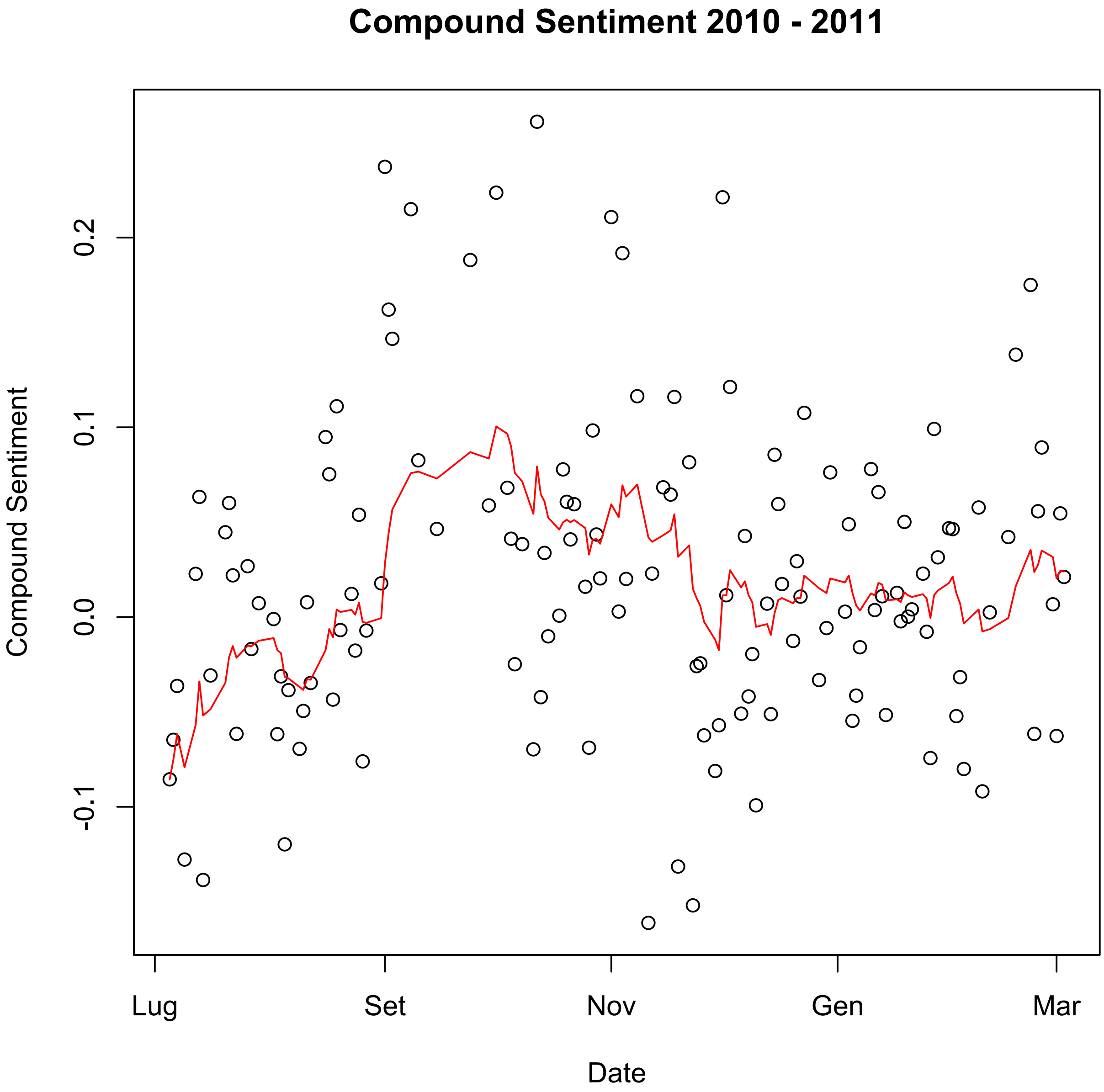
While choosing the initial estimates we can say $x_0 = 0$, but $P_0 = 0$ will cause the filter believe that $x_0 = 0$. By choosing any $P_0 \neq 0$ the filter will eventually converge.

Kalman Filter applications

100 obervations are generated using a noisy *sin* function, where noise is Gaussian. The state is then estimated using the Kalman Filter from the R package *FKF*.
The result is shown in the plot below



Since literature confirms sentiment noisy nature, we can believe that extracted sentiment is noisy, as VADER sentiment tool extracted scores already showed us. For testing purposes Kalman Filter has been applied to VADER sentiment compound scores.
The result is



It can be said that Kalman Filter provides an unobservable sentiment state estimate in the form of the unobservable process hidden behind sentiment scores. Under this point of view sentiment scores are seen as noisy measurements.
Given that sentiment scores are seen as noisy measurements of an unobservable state, the different opinions, namely the sentiment scores, are assumed to evolve around the unobservable sentiment state in a Gaussian way: this reflects the assumption of *white* noise in measurements.
Measurement noise normality is not a strong assumption since on one hand, given literature and VADER sentiment scores, it seems that sentiment is almost normally distributed around a

certain expected value, while on the other hand Kalman Filter seems to be quite robust to almost normal noise.

Conclusion

Given literature and evidences supporting sentiment noisy nature, Kalman Filter has been applied to VADER sentiment scores.
The result is a series of sentiment scores where the almost Gaussian measurement noise has been removed: such series reflects the unobservable sentiment state that generates the observed sentiment measurements, where observed sentiment measurement noise is just the outcome of the diverging opinions around the same fact.
Kalman Filter has been already used in previous papers regarding sentiment because unobservable sentiment series is easier to manipulate in order to study correlation and/or granger causality with other series.

Additional Research Topics:

- Kalman filter efficiency versus other smoothing algorithms.
  Kalman filter has been tested because it's the one used in other works regarding sentiment (Borovkova last paper for example). The *ratio* behind this could be the State Space Model framework which gives a qualitative interpretation to the procedure when applied to sentiment. However other algorithms may work even better.

- Kalman Filter sensitivity to outliers.
  VADER sentiment scores don't show many outliers, but such situation may change when new sentiment scores will be provided. Distortion caused by outliers has to be measured.