

Time Series Analysis: a case study

Manili, Maselli, Lete, Nicosanti, Trovarelli

4/27/2018

Activity tracking data: background and a priori reasoning

Any reasonable time series analysis must start by gaining some background knowledge about the dataset and the phenomenon at stake. Since we are reasonable analysts, we will not skip that step. Our investment in thinking about the dataset will be useful for the analysis and the modelling.

We consider a dataset made of 45 activities. Each activity corresponds to (part of) a bike ride, and is described by average **speed**, elevation (**elev**), and heart rate (**bpm**) measured at regular 30-second intervals, for a duration of 2 hours. Measurements are sampled with a smartphone, using some sensor device. The 3 time series are related by a common physical process: the bike ride.

Now, let us perform some intuitive reasoning on this physical process. We will base our *a priori*s on our cycling expertise.

Other things equal, bpm should be constant, especially in our case where it is an average of the bpm during 30 seconds, which creates a natural low-pass filter. The thing that makes increasing or decreasing the bpm should be interpreted, in that context, as the intensity of the effort done by the cyclist. Intuitively, speed and dfelev are related to the effort of the cyclist, and hence to bpm. It will be clearer *how* when we will define the utility function of the cyclist.

The speed series is a physical process mainly determined by the power that the cyclist the gravity. But in a recreational bike ride **gravity**, approximated by **dfelev** in our dataset, is likely to be the factor that dominates the fluctuations of **speed**.

Accelerations can be due to both **dfelev** or effort, variation of speed can be caused by many factors that might have different effects on bpm, in this sense, it seems difficult to use speed as a predictor for bpm. For example having to pass ahead of another cyclist may induce an acceleration using **power**, hence increasing the bpm, but accelerations due to losses in elevation reduce bpm as less effort is needed to maintain a given velocity. Thus it may not be a good predictor for bpm unless **dfelev** is controlled. Once dfelev is controlled, speed may capture at some extent the effort of the cyclist. But we have good reasons to think that variations of speed are caused mainly by other factors than **effort**: mainly by **dfelev**. Indeed, in a recreational bike ride it is not likely that a cyclist accelerate for the mere purpose of accelerating. Any cyclist, when travelling, want to maintain a stationary speed to minimize its efforts and maximize its utility, as a break implies a need of a futher acceleration, and accelerations use a lot of energy, especially when dfelev is positive. We could model the utility function of a cyclist as:

$$\begin{aligned} u_{i,t}(\text{speed}_t, \text{effort}_t) &= \text{speed}_t / \text{effort}_{i,t}(\text{speed}_t, \text{dfelev}_t) \\ \text{effort}_{i,t}(\text{speed}_t, \text{dfelev}_t) &= \sqrt{\text{speed}_t} * e^{\text{dfelev}_t} + 2\Delta\text{speed}_t * e^{\text{dfelev}_t} \end{aligned}$$

hence its utility function implies that at a given dfelev, a certain amount of effort is required to attain a certain speed, so that the equilibrium speed that maximizes the utility of the cyclist decreases during ascensions and increases during descents, as less effort is required to maintain a given speed. Moreover, he wants $\Delta\text{speed}_t = \text{speed}_t - \text{speed}_{t-1}$ to be zero at equilibrium. Thus, *assuming that our assumptions are correct*, the cyclist would like to maintain a stationary speed but variations in elevations makes the equilibrium speed fluctuate. For this reason we expect the variation of speed series mainly determined by **dfelev**: speed should be almost a negative mirror of dfelev. Though, we may have an unstable equilibrium speed during

descents, inducing more fluctuations of the speed, as the cyclist may want to break if the speed is too high (*tranne se si chiama Nibali*) or accelerate at some point as few effort is needed to increase the speed..

Last considerations regarding a regressive model: how the **bpm** series reacts during a bike ride is subjective to the peculiar individual and track. Though the sign of the correlation is not likely to vary accross tracks, it is likely that the size of the correlation vary, as how the bpm reacts to effort depends on a lot of context characteristics peculiar to the single track: different weather, temperature, environment, quality of the street etc.. Hence, it may not be recommended to pool tracks to estimate individual coefficients. Coefficients estimated through the single time series are likely to produce better forecasts than a pooling accross tracks, as they would be sensible to the particular context. Moreover, coefficients are likely to be not only heterogenous accross tracks but also accross time, suggesting that a non-linear modelling such as a dynamic linear regression state space model might really improve the specification. Since the effects may change due to past effort and shape of the path: not only tracks are heterogenous but also periods.

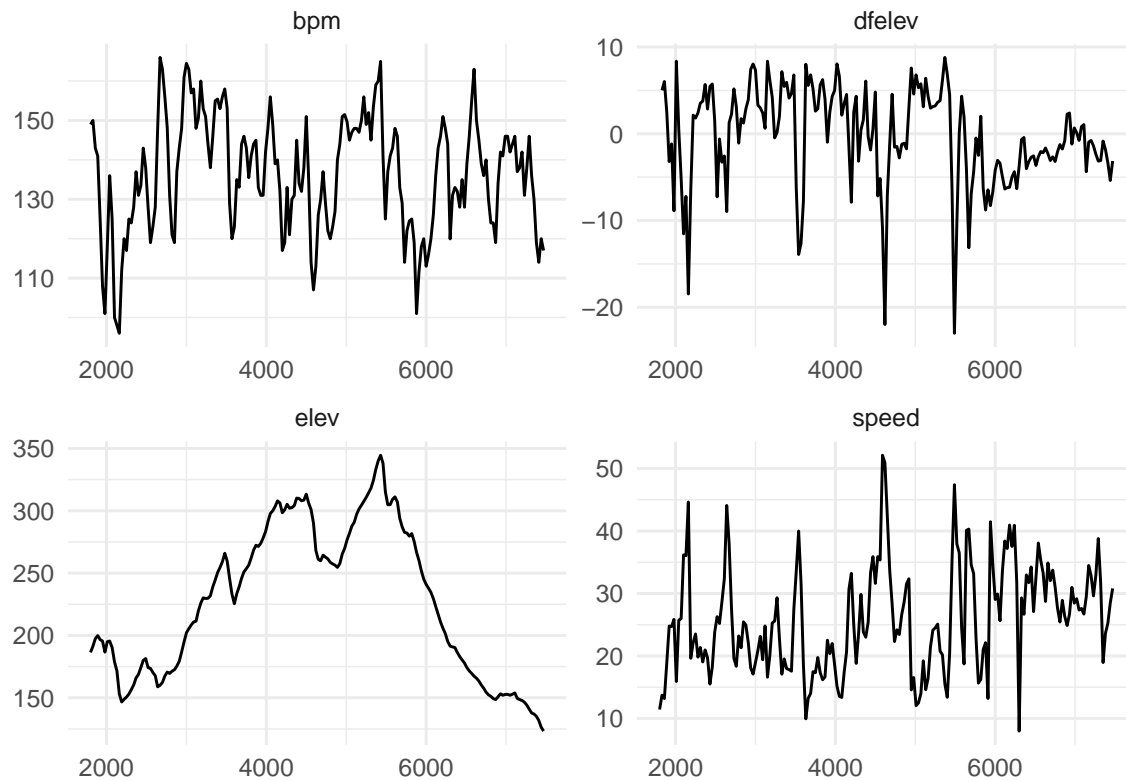
Have a nice reading,

Julien, Alfonso, Christophe, Andrea, Enrico

ARIMA models

1. Consider the first 95 minutes of activity track `track_id==38`. These correspond to the first 190 observations. Plot the data and describe what you see.

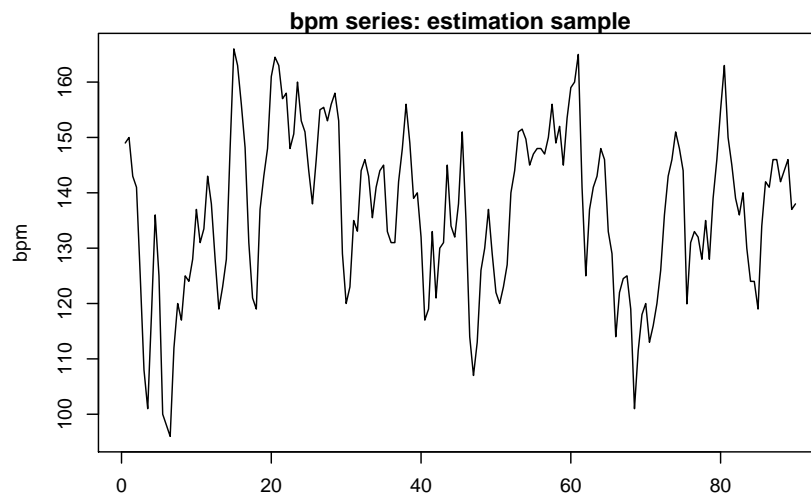
Track 38: first 95 minutes



From the plots we see a clear negative correlation between `speed`, and the `dfelev`, as well as a positive correlation between the `dfelev` and `bpm`. However, since there is a lot of variability in the elevation of the percouse, it is difficult to identify what are the effects of each variables on each other, since it is better when only one variable change at a time. However, we can say that `dfelev` dominates the other series, in the sense that it is this variable which has effects on `speed` and `bpm`. `bpm` is in some sense the consequence variable of `dfelev` and `speed`. `bpm` can measure the effort of the cyclist, but also the exhaustion of the cyclist, hence it can have negative or positive effects on speed in function of the situation.

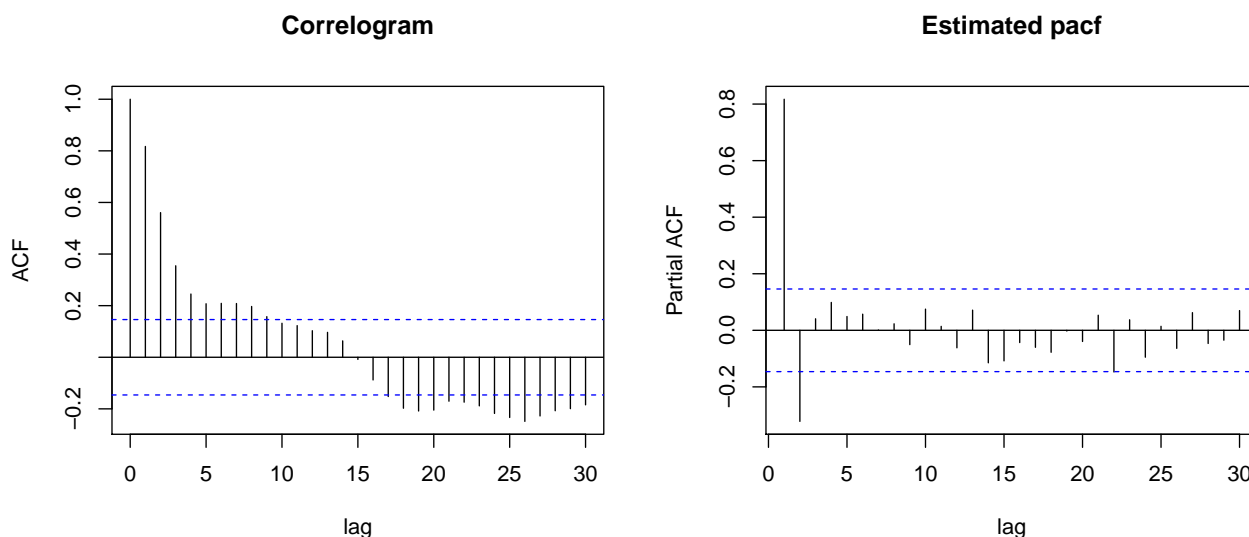
2. Use the Box-Jenkins approach to fit appropriate ARIMA(p,d,q) models to bpm. Compare alternative choices of (p,d,q) and finally choose a model.

We extract the first 90 minutes of the bpm series (estimation sample) and fit an ARIMA model using the Box-Jenkins approach. The first step is to plot the data in order to check for weak stationarity. Indeed, as ARMA models use the past to predict the future, it requires the future to be rather similar to the past, which technically translates into the assumption of second order stationarity of the underlying stochastic process.



The series looks stationary, we do not observe any particular trend, seasonality or discontinuities in the mean/variance. For natural reasons the data is bounded, as it is not possible to have 0 or 300 bpm (unless the cyclist pass out during the track). In addition of visual evidence, we ran an Augmented-Dickey-Fuller unit root test with drift, which also supports stationarity. Hence, we can reasonably assume that the underlying process is stationary.

Model specification step: let us plot the estimated acf and pacf to understand what is the temporal dependence structure of the series, from which we can guess which ARMA process is best suitable for our data.



It is apparent on the correlogram that we have a quite strong positive autoregressive component. Since we have significant negative lags after lag 15, we have at least two AR lags, or a combination of AR and MA components. The estimated pacf cuts after the first two lags, hence an AR(2) with a first positive coefficient and a second negative parameter could be consistent with the estimated dependence structure. The `auto.arima` function proposes an ARMA(1,2) model. We check also for an ARMA(2,1), for the sake

of prudence. Hence we compare the following models: AR(2), ARMA(2,1), ARMA(1,2). We fit the three models and perform a diagnostic on residuals to check whether the assumptions of the model are met, and evaluate which model performs best using some statistics. The resulting comparison is the following:

All three models passed the diagnostic checking, i.e. from the acf and Ljung-Box tests: no evidence of significant autocorrelation between residuals. We discard ARMA(2,1) as estimated coefficients have quite high standard errors, and MA[1] and AR[2] coefficients are not even significant. So we have to choose between the ARMA(1,2) and the AR(2) model. We notice that all models are wrong, including these two, hence there is no risk of discriminating the true model.

The ARMA(1,2) model performs better in terms of the p-values of the Box-Pierce tests (which are almost 1 at all max.lag joint significance tests) and in terms of log-likelihood, but standard errors of its MLEs are quite higher than those of the AR(2) model, and the second MA component is borderline significant. A reason that may induce us to choose the AR(2) model is much simpler to estimate, as the presence MA components requires to use more sophisticated and hence less precise estimation methods. Moreover, the principle of parsimony prescribes to select the *simplest model acceptable*. We should inflict a penalty to the ARMA(1,2) model, as increasing the number of parameters generally improve the model fit, but can also create overfitting and make forecasts more unstable.

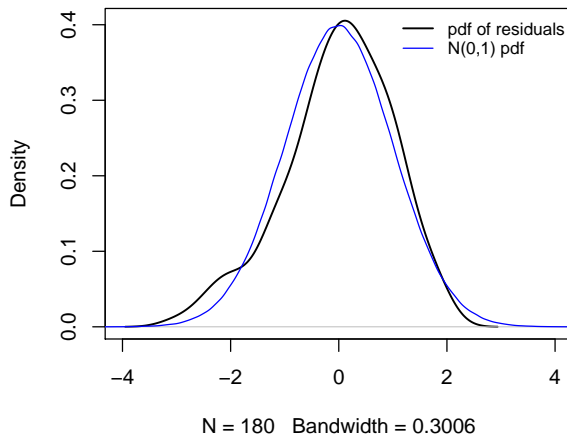
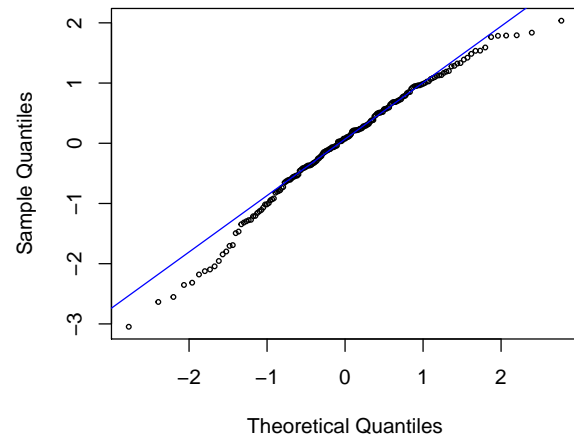
More formally, we can check whether model selection criteria provide the same judgment. Information criteria integrate the trade-off between model fit, measured by $-2\loglik(MLE)$, and overfitting including a penalty to model complexity, measured by the number of parameters to estimate. According to Akaike, the penalized $-2\loglik$ of AR(2) is smaller than the one of the ARMA(1,2), supporting the AR(2) model. However, statisticians generally prefer the BIC criterion (because it is Bayesian, so it is nicer by definition). Since AIC generally inflicts the least penalty to the number of parameters among ICs, BIC and HQC should also support the AR(2) model. Indeed, given the sample size of $T = 180$ observations, $\log(T) = \log(180) = 5.20 > 2$ and $2\log(\log(T)) = 2 * 1.65 = 3.30 > 2$, so that the Schwarz' BIC and the Hannann-Quinn IC select even more strongly the AR(2) model. Hence, we choose the AR(2) model.

```
##
## Call:
## arima(x = track_bpm, order = c(2, 0, 0))
##
## Coefficients:
##          ar1      ar2  intercept
##       1.0784 -0.3201   136.8531
## s.e.  0.0702  0.0703    2.4402
##
## sigma^2 estimated as 63.82:  log likelihood = -630.11,  aic = 1268.21
```

The estimated model variance is 63.41. All coefficients are strongly significant. The estimated model is the following:

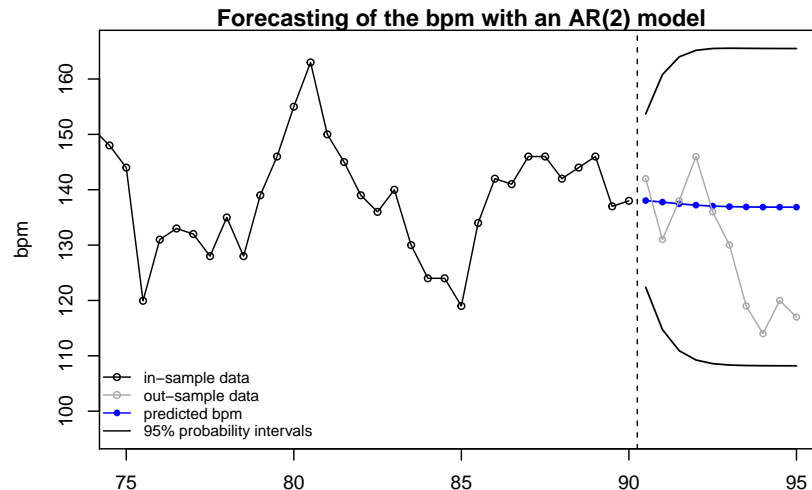
$$Y_t = 137 + 1.08Y_{t-1} - 0.32Y_{t-2} + \epsilon_t, \quad \epsilon_t \stackrel{iid}{\sim} WN(0, 64)$$

Before going further, we perform a quick normality test on residuals, as we will be induced to use prediction intervals that are based on a normality assumption in subsequent questions.

Estimated density of the standardized residuals**Normal Q-Q Plot**

It looks quite normal, though it is not perfect. Also, the Kolmogorov-Smirnov normality test accepts the null of normality with a quite high p-value of 0.7122. Hence our credible intervals will be more or less credible.

3. Forecast bpm for the 5 final minutes. Plot the forecast, along with confidence intervals. Compare them to the actual observed values. What do you see? Comment.



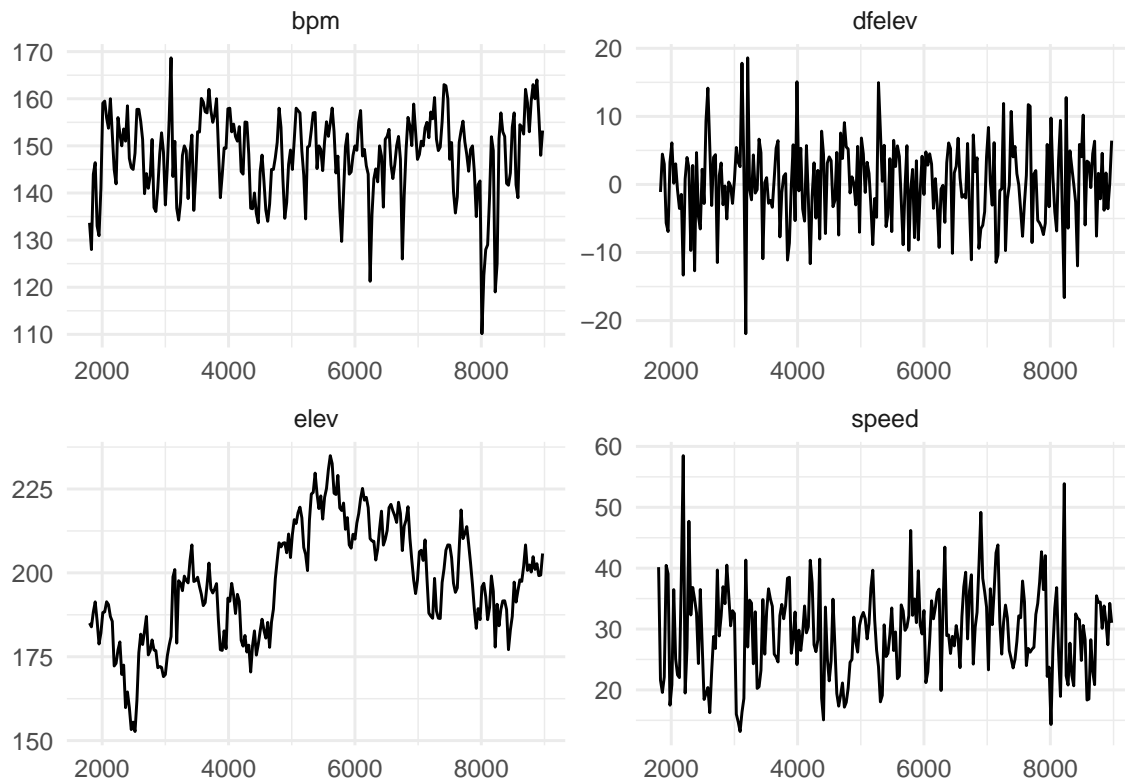
Dynamic predictions quickly converges to the intercept. The first six predictions are quite good compared with the actual realizations, but there is a progressive change (decrease) in level, so that we are systematically overpredicting after 5 steps ahead, though the observations remain in the forecast 95% probability intervals. The problem of the model is that it cannot predict these changes in level, we would need to enrich the model with covariates in order to be able to predict these changes. We can conclude that the model provides reasonable short-term forecasts, but one has to be cautious when it comes to forecasting at more than 2 minutes ahead.

Dynamic Linear Models

Track 28

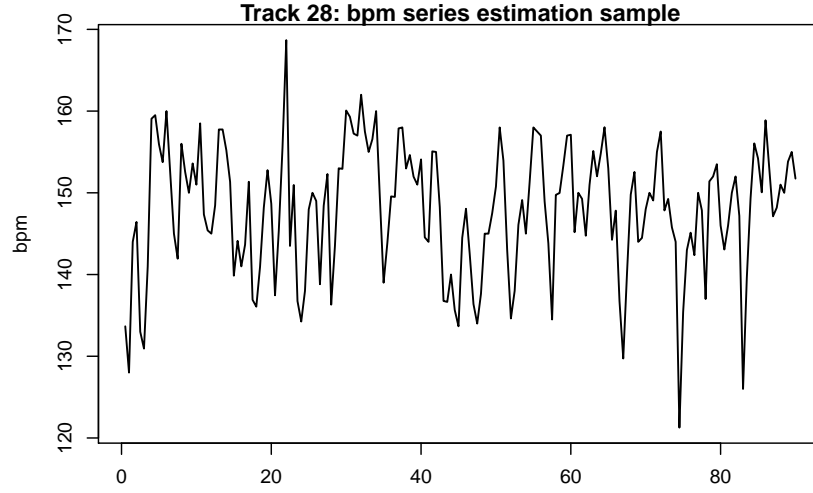
1. Plot the data from `track_id==28`. Very briefly describe what you see.

Track 28: first 95 minutes

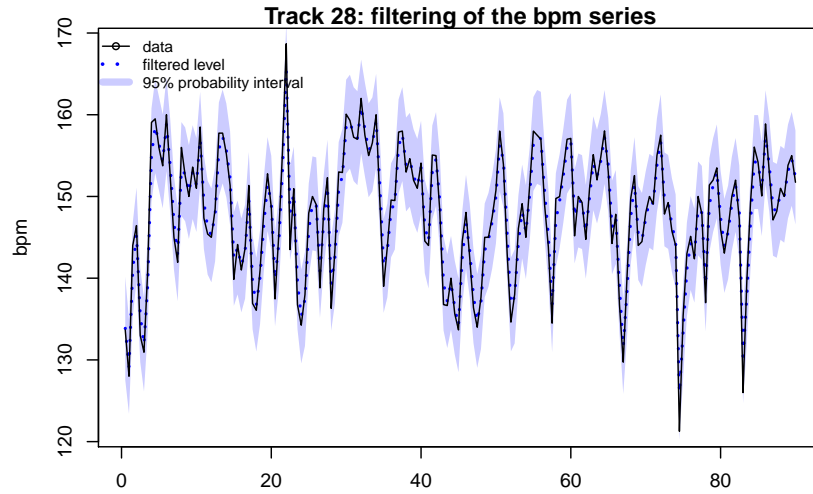


We have a pretty noisy track, with a lot of small elevations and small descents. The elevation series is very similar to a random walk, and indeed, when we look at the differenced elevation series, it is behaved as a white noise. As a consequence of the random walk behavior of elevation, we have for the bpm and the speed series a mean level around which observations fluctuate. If we were to consider the bpm series alone, we could model the series with a random walk plus noise state space process, i.e. we could see the bpm series as the observations of a level (the true bpm) plus some measurement error. As the bpm series does not exhibit any trend or seasonality, a constant forecast function seems reasonable.

2. Consider the first 95 minutes. Using the first 90 minutes, build a random walk plus noise model for bpm (use $m_0 = 140$, $V = 10.77$, $W = 34$). Plot the filtering estimates with 95% credible intervals.



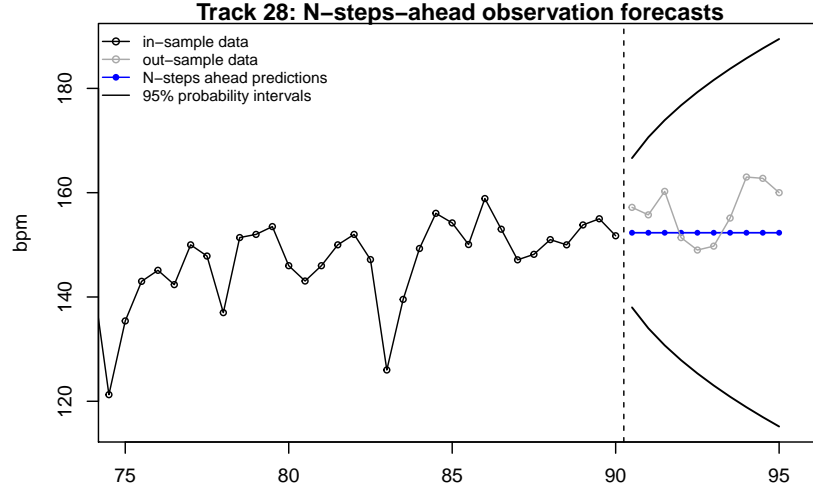
We specify the local level model prescribed in the question using the package `d1m`. Then we apply the Kalman filter using this model on the `bpm_28` series and plot the filtering estimates together with their 95% probability intervals. The filtering distributions, i.e. the distributions of $\theta_t|y_{1:t}$, are normal with mean m_t and variance C_t , hence we can easily construct confidence intervals for the filtering estimates.



It is apparent from the time plot that filtering estimates essentially coincide with actual observations. This is due to a high signal-to-noise ratio σ_W^2/σ_V^2 , which is more than 3 in our case. This implies that the updating rule assigns an important weight to the current observation. And this is reasonable given the relatively high evolution variance compared to the observation variance: if we see a large deviation from our previous estimate of the level, we would attribute this to a change in the level θ_t rather than to a measurement error.

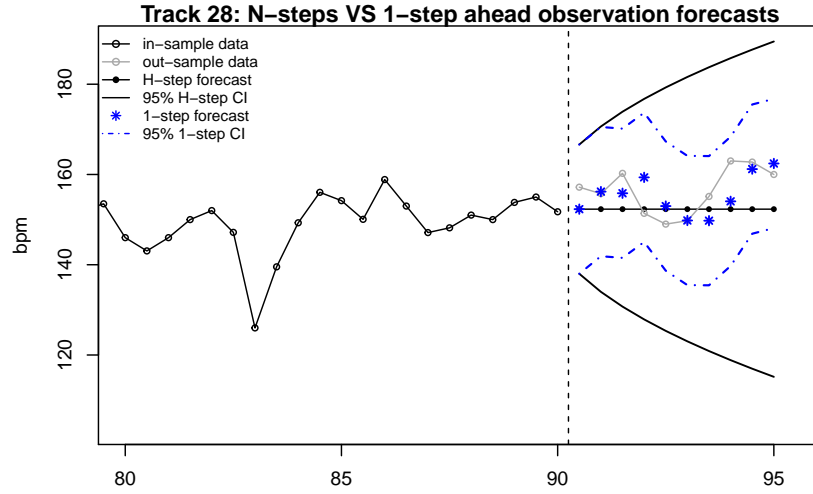
Regarding the assumptions of the model on the innovation process, as reasonable economists, we assume that assumptions hold. Hence, $\frac{e_t}{Q_t} \stackrel{iid}{\sim} N(0, 1)$, where $e_t = y_t - f_t$.

3. Use data from 1 to 90 minutes to compute the forecasts for $t + 1, \dots, t + 10$, corresponding to the last 5 minutes. Plot them with credible intervals. Compare them with the 1-step ahead forecasts for the same period, after plotting them with intervals, too.



We observe that the N-steps-ahead forecasts are constant throughout the 10 steps, i.e. we have a constant forecast function: as expected since the random walk plus noise is a polynomial DLM of order one. Nevertheless, our 10-step-ahead forecasts appear to have performed quite well with respect to realized observations, all observations remained inside the 95% prediction interval.

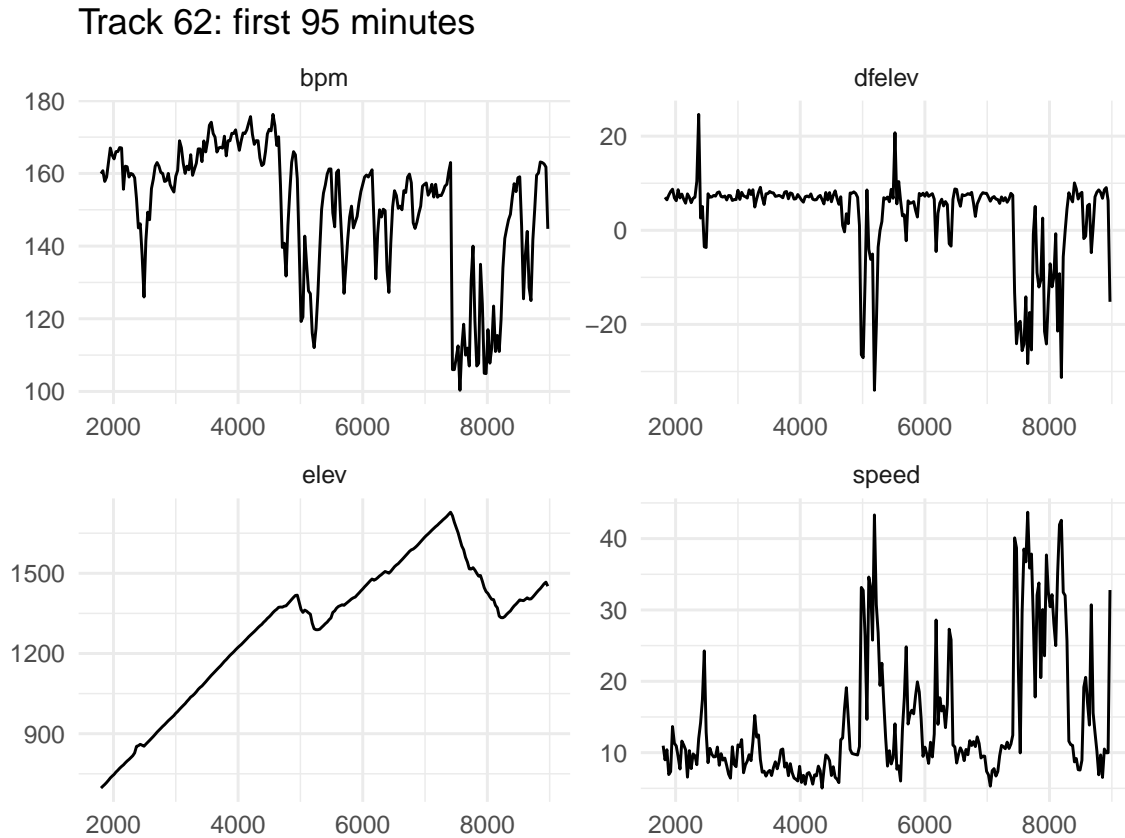
Now let us compare the N-steps-ahead forecasts with the 1-step-ahead forecasts:



Naturally, N-step ahead forecasts becomes more uncertain than 1-step-ahead forecasts as the forecast horizon increases, summarized by an enlarging prediction interval. On the other hand, the range of one-step-ahead forecast credible interval remain constant, since we are sequentially updating the filtering distribution, incorporating the new observed information y_{T+h} . We see that at the first 5 steps, dynamic and static forecasts are close, but then diverge as the estimated bpm level of θ_{T+h} changed.

Track 62

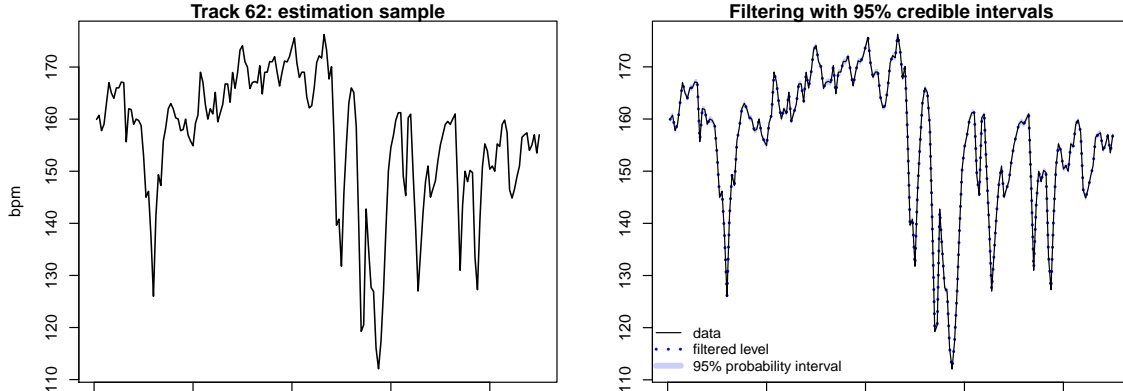
1. Plot the data from track_id==62. Very briefly describe what you see.



Unlike track 28, this journey is characterized by a steady elevation series, it seems to be a mountain journey, with constant ascensions and descents. The consequence of the steadiness of the elevation series, is that the other variables such as speed and bpm are non stationary, as descents and elevations imply different local levels with noise of the variables. During ascensions, the level of bpm is high with low noise, while during descents, the level of bpm is low with medium noise. For all these reasons, without employing formal stationarity tests, we can safely state that all these four series are non-stationary. Hence ARIMA models are not appropriate to model these series, we may want to model explicitly the non-stationarity of the series. State space models, which do not require stationarity, could capture the main changes in level and other minor changes of the series. We could even use a linear growth model for the bpm series as we see that there is a growth of the bpm during ascensions.

2. Consider the first 95 minutes. Using the first 90 minutes, build a random walk plus noise model for bpm (use $m_0 = 140$, $V = 0.1$, $W = 10$). Plot the filtering estimates with 95% credible intervals.

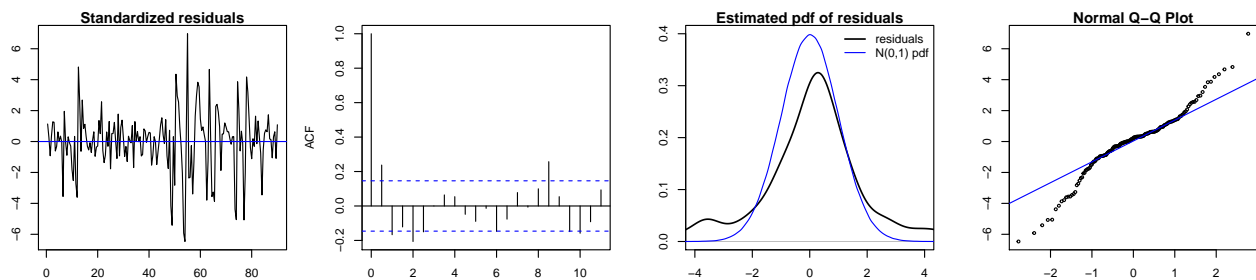
As before, we construct the local level model prescribed in the question using the package `d1m`. Then we filter the `bpm_62` series with this model and plot the filtering estimates with their 95% probability intervals.



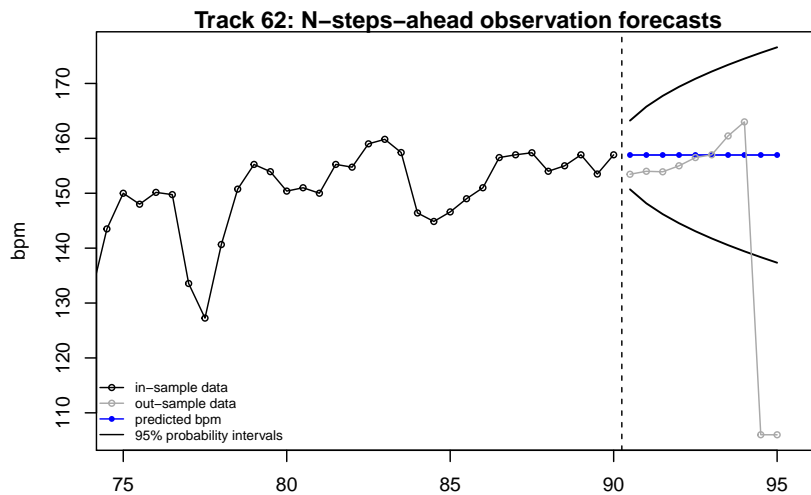
Due to an extremely high signal-to-noise ratio ($\sigma_W^2/\sigma_V^2 = 10/0.1 = 100$), Kalman filter basically estimates the “hidden” level with the current observation. Such a high SNR takes away the interpretation as “hidden state” and makes the filtering somehow useless, as it does not filter anything (it should filter the noise from the signal).

The given values of σ_V^2 and σ_W^2 have been probably obtained with an MLE estimation considering this single track. In fact, we can see V as the precision of the heart-rate monitor (technically, it is a 30s average of measurements). Such a low observation variance signals that something is going wrong, as Michele uses a cheap heart-rate-monitor (a chest band), which is hard to believe so precise. It may be an artifact of calculating the MLE considering this single track. Moreover, though we expect σ_W^2 to vary along tracks, as the evolution pattern of the “true” bpm process (our unobservable state process) depends intuitively on the peculiar track, a varying observation variance suggests that there is a problem of identifiability between the two variances. The variance of the measurement error σ_V^2 should be roughly constant along tracks, as it is the same Michele with the same measurement device in all tracks.

Regarding the assumption of the model, we can easily investigate on the innovation process, if assumptions are correct it should be $e_t \stackrel{iid}{\sim} N(0, Q_t)$. We report that the standardized innovation process exhibits short term correlation, is non-stationary in the variance, and have larger tails than a $N(0, 1)$.

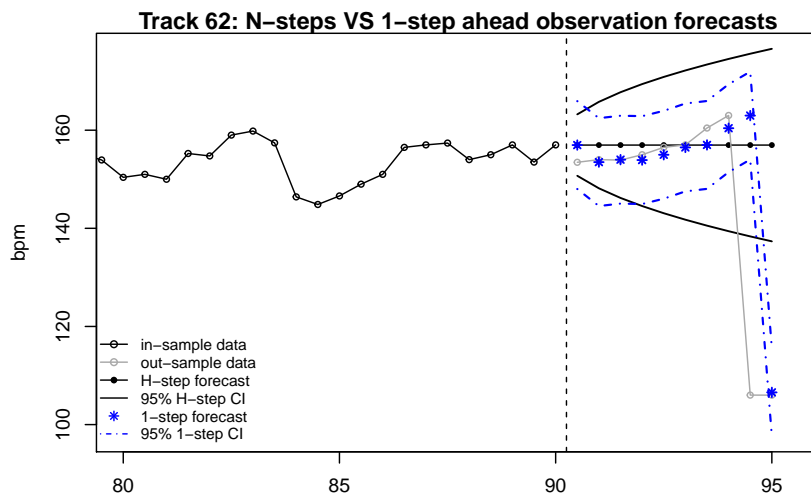


2. Use data from 1 to 90 minutes to compute the forecasts for $t + 1, \dots, t + 10$, corresponding to the last 5 minutes. Plot them with credible intervals. Compare them with the 1-step ahead forecasts for the same period, after plotting them with intervals, too.



We see that forecasts are quite good (until the '94 crisis). The key observation is that we have a non-stationary series that can experience major change points in level. We see that at the end, forecasts completely failed because there have been a drastic change point in the level of bpm. Though, we cannot do better without covariates, we can only pray that there will be no change point. With `dfelev` as a covariate, we could have predicted ex-ante this radical change in the level of bpm, as it is due to an extreme value of `dfelev` which corresponds to a change in a local trend of `elev` (the peak). Of course all this provided that we have the N-step-ahead values of the covariates. If we do not have this data, it is still difficult to predict change points, because they are generally due to a sudden change in covariates that are themselves unpredictable. Hence, generally speaking, one must be cautious when predicting too far away with a local level model, as the level is... local. Especially with series showing non-stationarities. Univariate models generally produce reasonable short term forecasts, but as we extend the prediction horizon, the risk of a drastic change point that would make fail forecasts increases.

Now let us compare the N-steps-ahead forecasts with the 1-step-ahead forecasts:



We see that the one-step-ahead forecasts successfully adapted to the change point after the first big fail of the forecast. Which is the best we can do without predictors: quickly adapt.

Regression Models

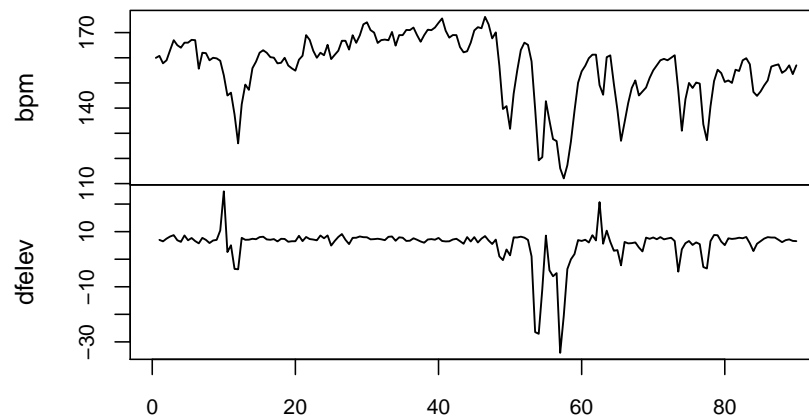
Static regression

We have seen in the previous questions that univariate approaches may fall short in terms of predictive performance when a time series experiences change points or non-stationarities. We now consider **bpm** and **dfelev** from **track_id==62**. As a first step, we fit a static regression model, then from residual analysis we will understand how to improve the specification. The static model is:

$$Y_t = \beta_0 + \beta_1 x_t + \epsilon_t$$

Where Y_t is **bpm**, x_t is **dfelev** and $\epsilon_t \stackrel{iid}{\sim} N(0, \sigma^2)$ are iid measurement errors.

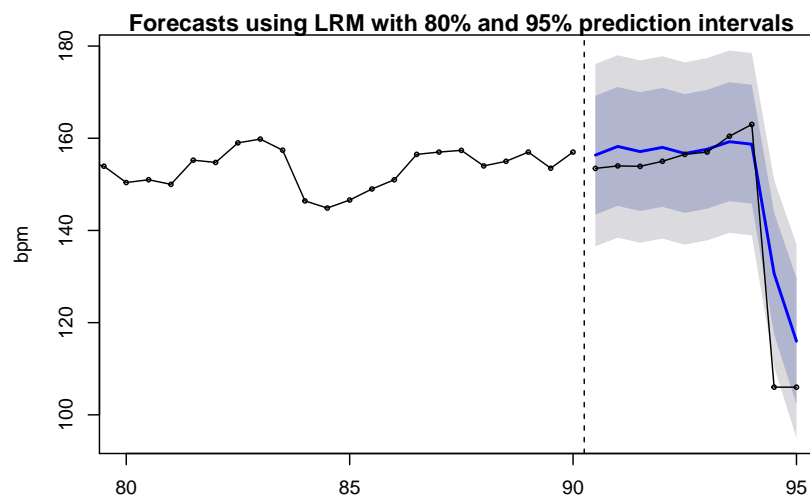
We take the first 180 observations for estimation and next 10 observation for forecasting exercise. The estimation sample is the following:



We fit the regression using **lm** and obtain the following results:

```
##      constant  dfelev
##  coeff  148.500   1.346
##    se     1.020   0.123
```

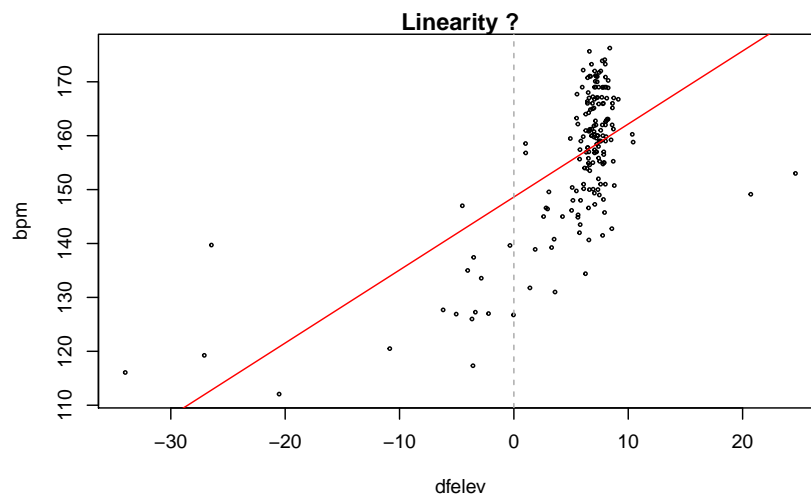
The estimated residual standard error is 10.15, and the adjusted R^2 is 0.41. Now we forecast the next 10 observations with the function **forecast** (package **forecast**):



Without computing forecast performance indicators, we see a clear improvement of prediction performance with respect to univariate forecasting using the random walk plus noise model. The **dfelev** series provides a

lot of information about the “intensity of the effort of the cyclist” process that drives bpm. It has clearly a “leading indicator property”, allowing to predict a change point in advance. Indeed, we observe that the model successfully predicted the change point that has occurred in '94, where univariate models failed.

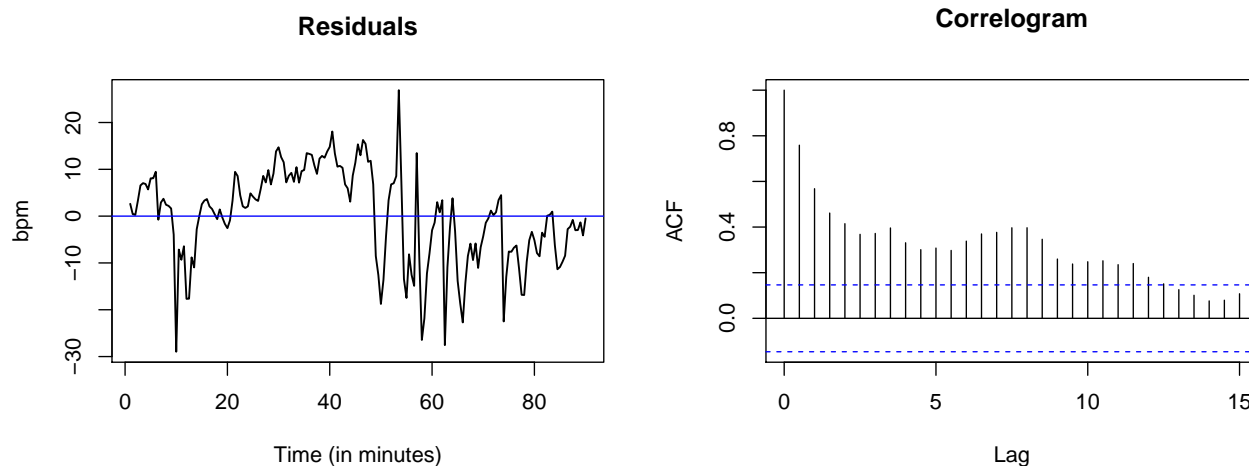
Now, let us focus the analysis on the model. From the interpretative point of view, the model ignores the temporal dimension and simply apply OLS. Hence, the estimated marginal effect of dfelev is linear and constant over time, and the constant is constant. The model simply fit a line with least squares, as show the following graph:



The static regression, given its forecast performance, is a reasonable starting point as elevation gains and loss are powerful predictors for variations of bpm, but we see that from the modeling standpoint it needs some improvements.

Remaining in the static point of view, we see that the relationship between bpm and dfelev is not uniform through the whole series, from -30 to 0 the slope is positive but weaker than when dfelev is positive, and becomes sharper as dfelev becomes positive. Including a dummy for negative and positive dfelev, and eliminating the two crazy outliers in the positive dfelev side, we may do quite better. Without the dummies, we see that this non-linearity creates overpredictions when dfelev is negative and underpredictions when dfelev is positive (i.e. biased forecasts).

Moreover, OLS assumes stable parameters over time, but looking at residuals we remark that the temporal dimension also matters. Let us have a closer look to the residuals:



Without employing any formal test, it is apparent on the plot that residuals are not random and highly

positively correlated over time, strong signal of parameter instability. The model is systematically underpredicting in the first part of the series, and then overpredicting. Hence the assumption of stability of parameters over time is likely to fail.

We also note that though the dfelev variable is fairly constant during the first part of the series, bpm increases, suggesting that either the intercept or the marginal effect of dfelev vary over time. Intuitively, dfelev has less effect at the beginning than at the end of the track if we think about an accumulating fatigue. Hence, a more sensible specification allowing for parameters to change over time may be able to improve these results and would be more realistic from the philosophical point of view, as assuming a constant linear effect of dfelev remains a very rough approximation.

Dynamic Linear Regression Model

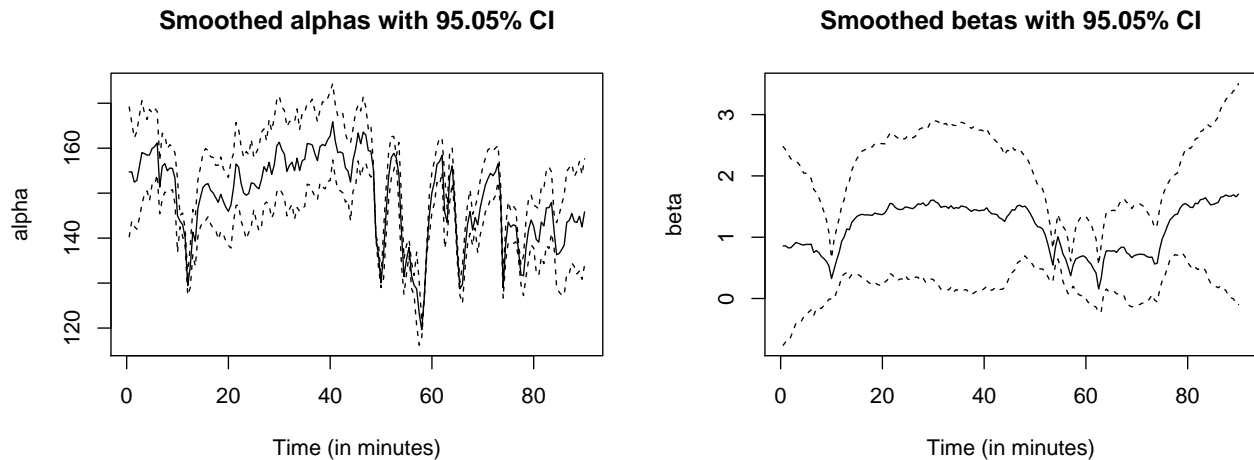
Let us see what can do a DLM regression allowing parameters to change over time, formally:

$$\begin{aligned} Y_t &= \alpha_t + x_t \beta_t + \epsilon_t \\ \alpha_t &= \alpha_{t-1} + w_{1,t} \\ \beta_t &= \beta_{t-1} + w_{2,t} \end{aligned}$$

Where Y_t 's are iid measurement, $\epsilon_t \stackrel{iid}{\sim} N(0, \sigma_\epsilon^2)$, $w_t \stackrel{iid}{\sim} N(0, W)$, W is a constant diagonal matrix, i.e. the state vector θ_t is composed of two independent random walks. We estimate the model variances with the MLE, then we use the DLM to derive the smoothed state estimates.

```
##          dV      dW1    dW2 Converg.
## par 0.0006 22.7981 0.0457         0
## se  0.0304  2.9057 0.0217        NA
```

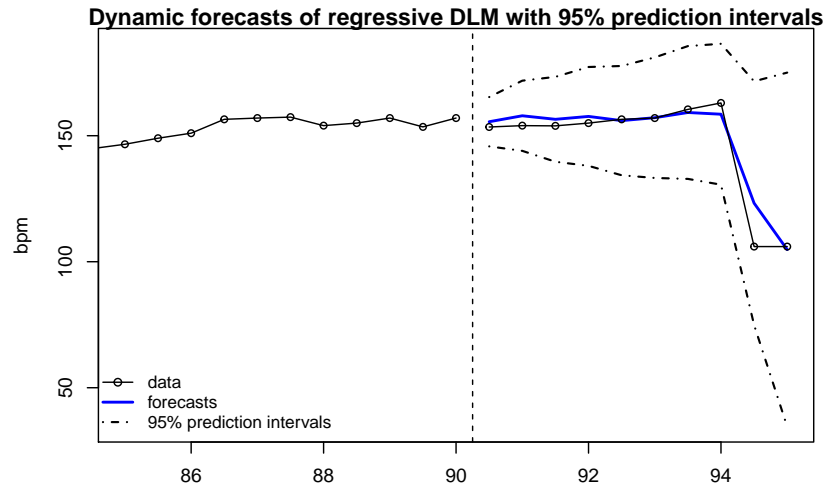
We note that V seems excessively tiny given the cheap measurement device. Also, MLEs change a lot according to the initial values we set, sometimes not converging numerically: MLEs are unstable. Hence, we may want to assume a constant intercept, letting only β to vary over time, in order to increase stability. But let us continue the modelling with both parameters changing over time.



Interesting is the history of β : the marginal effect of dfelev is greater during ascensions than during descents. It is apparent on the plot that there are two “levels” of β s: one for descents and one for ascensions, which is consistent with our previous observations regarding the non-stability of effects of dfelev accross neg/pos dfelev and time. During ascensions smoothed estimates carry much more uncertainty than in descents, summarized by larger confidence intervals (going from 0.5 to 3). Regarding the α process (level), it tends to increase during prolonged ascensions, and decrease during descents, again consistent with our previous reasoning saying that the intercept may vary over time, especially during constant elevation

gains because of an accumulating fatigue. At the end of the series alpha becomes very volatile as there is more volatility in elevation gains and losses, making varying the intercept in a non-regular way.

Now, let us forecast the next 10 observations, to see what this model can do.



Forecasts are excellent. The model performed much better than the static regression in terms of predictive performance, both before and during the change point. Using the MAPE, we have a mean absolute forecast percentage error of 2.86% for the dynamic model, and 4.48% for the static model, which represent an improvement of -36% of MAPE.

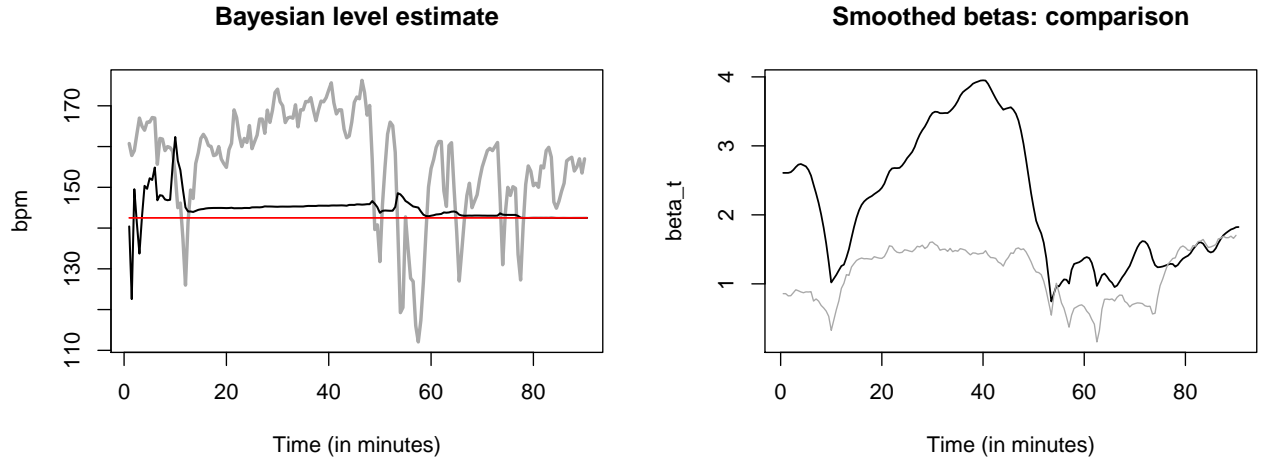
Dynamic Linear Regression Model with Pooled MLEs

We have the same DLM as before: dynamic linear regression model, with the regression parameters as state process, and in this case a zero evolution of the level (i.e. a constant). The main difference with before is that the MLEs of the two remaining variances (observation and evolution of the beta coefficient) are computed using the information of all tracks: the MLEs are found finding the value of the two variances that maximizes the *sum* of the likelihoods of all tracks. This is an intelligent way to exploit the commonality of the series: the observation variance. As all tracks are recorded with the same device, they should have the same measurement variance. Hence, it allows to partially overcome the problem that we encountered before: unstable MLEs and varying observation variances accross tracks. The homogenous MLE for V is a little bit higher (more reasonable) than for the non-pooled model: we pass from $V = 0.0006$ to $V = 103.2$ (but this is also due to the fact that before we allowed the constant to vary). $dW_2 = 0.1$.

The drawback is that while observation variances should be constant over tracks, the variance of the betas instead should vary, as the change in the marginal effect of *dfelev* depends on variables peculiar to the single track (type of journey, mountain or city, wind, temperature, quality of the street, altitude, whether the cyclist ate sausages or whipped cream at breakfast etc.). The model assumes homogeneity accross tracks of *both* the observation variance and the evolution variance. However, we notice that it is still better than before, because we were not able to identify correctly the variance of the betas in the single track anyway, so at least we do one half better.

We can still imagine something like doing this procedure to extract the homogenous observation variance, and then do another optimization procedure, taking the homogenous observation variance as given, to find the heterogenous MLE of the evolution variance using the single track. But as noted Michele, it is not so simple, and it might require more complex estimation methods.

Regarding the assumed constant level, though the level may vary over time, it seems hard to pool the level variances among all series, as its variance depend even more on the peculiar track. Moreover, it is an useful simplification to have more precise MLEs, as distinguishing between 3 variances may not be manageable.



A rapid comment on these plots: since the level is constant but random, in the sense that it is unknown, the smoothing of the constant represent a Bayesian estimate: $E(\alpha|y_{1:T})$. Instead, filtering estimates represent the Bayesian learning process of the constant, they vary not because of the evolution of the level but because we adjust the estimate as we receive new information, it follows that they eventually coincide with the Bayesian estimate. The smoothing is naturally constant as it retraces the past history of a constant with the information of the whole sample, and by definition, the past history of a constant is constant. However, if we have new information, the Bayesian estimate may still change. The Bayesian estimate is constant only retrospectively, not forwardly.

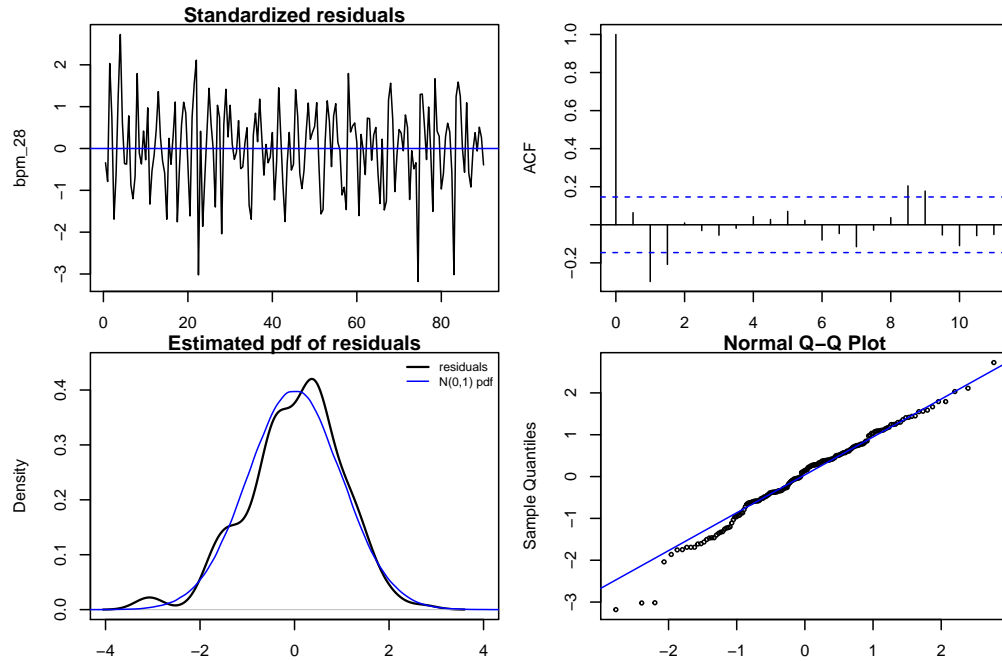
Regarding the smoothed betas, we see that they tend to increase during prolonged ascensions, and decreases after descensions. It seems reasonable. After the descent, the beta restart at a lower level, suggesting that the ascension strengthened the cyclist, and is less affected by ascensions after having rested a little bit during a descent. We notice that we have a quite different picture than with the non-pooled model, also because the non-pooled model included a varying level. However, we see that the trends are quite the same, and at the end of the series they essentially coincide.

Appendix: some curiosities for the interested reader

Track 28: Diagnostic checking of the fitted local level model

“Regarding the assumptions of the model on the innovation process, as reasonable economists, we assume that assumptions hold. Hence, $\frac{e_t}{Q_t} \sim iidN(0, 1)$, where $e_t = y_t - f_t$.”

Actually we were not so reasonable, because we performed a quick check on the properties of the innovation process: if the assumptions of the model are correct, the standardized innovation process should be iid $N(0, 1)$. Let us investigate whether the assumptions of the model are met:



```
## [1] 0.01578668
```

```
## [1] 0.9976932
```

We observe a slight short term correlation between residuals, this may be due to the few outliers visible on the plot of residuals. Errors change frequently of sign, are distributed around 0 and have a standard error or 1, and seems normally distributed. Summing up, evidence supports that the model is correct.