Time Series Simulation

Rachael Phillips

In this project, we will be in a setting of multi-person multi-dimensional time series. We will use an online SL and can incorporate learners which pool across subjects and individualize subjects (via more aggressive modeling of the effect of the baseline covariates). The online cross-validated risk is individualized, but we will average across subjects because it's a more interpretable measure.

Simulation Overview

Simplistic settings which can capture the following:

- A situation where subjects are iid (baseline cov variation is null)
- Time series variation is very much a function of baseline covariates
- Time series variation is very much a function of baseline covariates that you don't measure (unexplained heterogeneity).
- Across time the process is stationary
- Time is not stationary (sudden jumps) so you cannot learn from the past how much it will change in the future.

Previously I simulated from relatively simplistic ARIMA models.

Today I model the continuous MIMIC outcome data with ARIMA.

Simple ARIMA Simulations

ARIMA Introduction

An auto-regressive integrated moving average model (ARIMA) is specified by three order parameters: (p, d, q).

p is the number of autoregressive terms The p is the auto-regressive (AR(p)) component and refers to the use of past values in the regression equation for the series. The auto-regressive parameter p specifies the number of lags used in the model. Intuitively, this would be similar to stating that it is likely to be warm tomorrow if it has been warm the past p days.

d is the number of nonseasonal differences The d represents the degree of differencing in the integrated (I(d)) component. Differencing a series involves subtracting its current and previous values d times. Often, differencing is used to stabilize the series when the stationarity assumption is not met. Intuitively, this would be similar to stating that it is likely to be same temperature tomorrow if the difference in temperature in the last d days has been very small.

q is the number of moving-averages terms A moving average (MA(q)) component represents the error of the model as a combination of previous error terms, where q defines the number of terms to include in the model.

Differencing, autoregressive, and moving average components make up a non-seasonal ARIMA model which can be written as a linear equation:

$$Y_t = c + \phi_1 y_{dt-1} + \phi_p y_{dt-p} + ... + \theta_1 e_{t-1} + \theta_q e_{t-q} + e_t$$

where y_d is Y differenced d times and c is a constant.

ARIMA models can be also specified through a seasonal structure. In this case, the model is specified by two sets of order parameters: (p, d, q) as described above and $(P, D, Q)\_m$ parameters describing the seasonal component of m periods.

ARIMA methodology does have its limitations. These models directly rely on past values, and therefore work best on long and stable series. Also note that ARIMA simply approximates historical patterns and therefore does not aim to explain the structure of the underlying data mechanism.

Resources

- A Short Introduction to ARIMA
- Time Series: AR, MA, ARMA, ARIMA
- Hyndman and Athanasopoulos Forecasting: Principles and Practice

ARIMA Simulations with White Noise

White noise time series can be useful because the stochastic behavior of all time series can be explained in terms of the white noise model. We simulate Gaussian white noise, wherein $wt \sim_{iid} N(\mu, \sigma)$. μ is set to 0 or a baseline covariate value and σ is set to 1 or a baseline covariate value. We consider three scenarios:

1. μ is 0 and σ is 1, the time series is a function unexplained by baseline covariates.
2. μ depends on a baseline covariate and σ is 1, the time series is partially a function of baseline covariates.
3. μ and σ depend on a baseline covariate values, the time series variation is a function of baseline covariates.

For each of the three scenarios, we simulate $N = 500$ subjects each with $n = 1000$ observations from the following models:

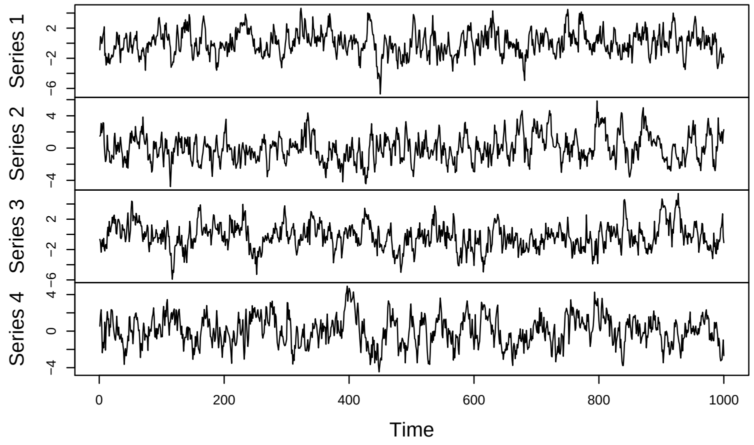
1. An autoregressive model of order 1 ($p = 1$), where each value of y equals the previous value times 0.8, plus the white noise.
2. A moving average of order 1 ($q = 1$), where each value of y equals the latest bit of white noise, plus 0.8 times the previous value of white noise.
3. An autoregressive moving average model of order (1, 1), combining the two above.
4. An ARIMA(1, 1, 1) model that is the cumulative sum of the ARMA(1, 1), so the first difference of the time series is stationary.

```r
# mu is 0 and sigma is 1
wt_ts_0 <- sim_wt_ts()

# mu is W3 and sigma is 1
W1 <- runif(500, min = -1, max = 1)
W2 <- rbinom(500, prob = plogis(W1), size = 1)
W3 <- W1 + W2
wt_ts_W <- sim_wt_ts(mu = W3)

# mu is W3 and sigma is W4
W4 <- W1 + 1
wt_ts_WW <- sim_wt_ts(mu = W3, sigma = W4)
```

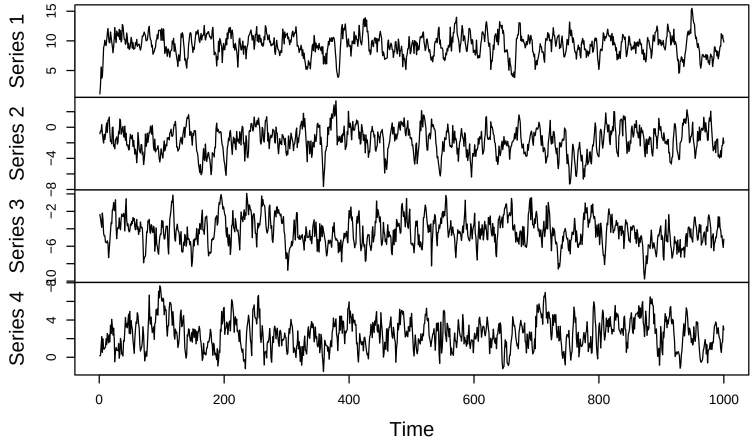
Now we can visualize the various models.

1. An autoregressive model of order 1 ($p = 1$), where each value of y equals the previous value times 0.8, plus the white noise.

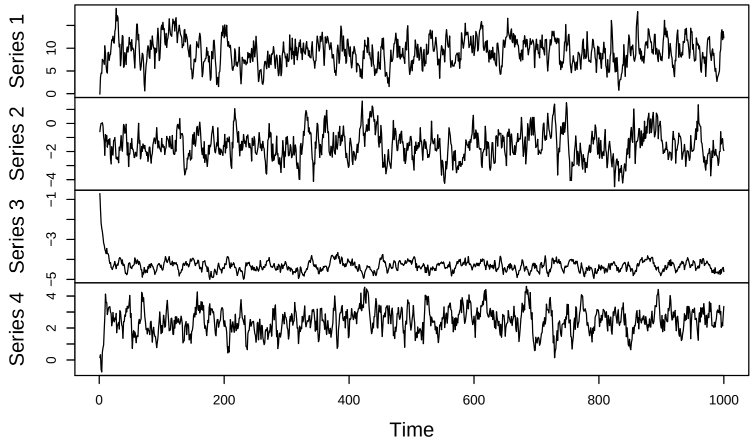
AR(1): μ = 0, σ = 1



AR(1): μ = ϕ(W), σ = 1

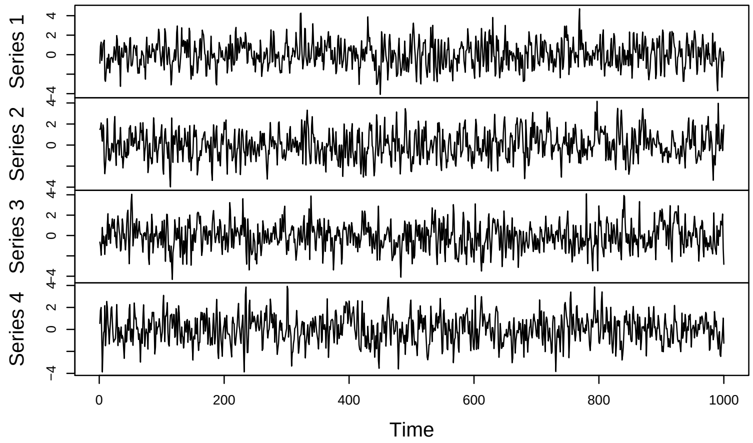


AR(1): μ = ϕ(W), σ = ϕ(W)

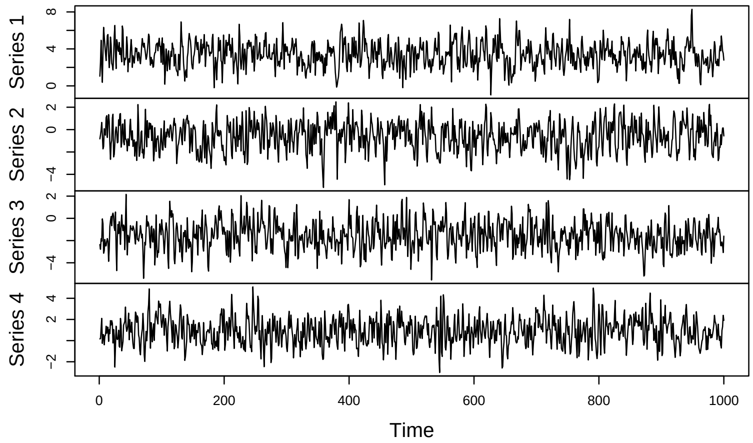


2. A moving average of order 1 ($q = 1$), where each value of y equals the latest bit of white noise, plus 0.8 times the previous value of white noise.

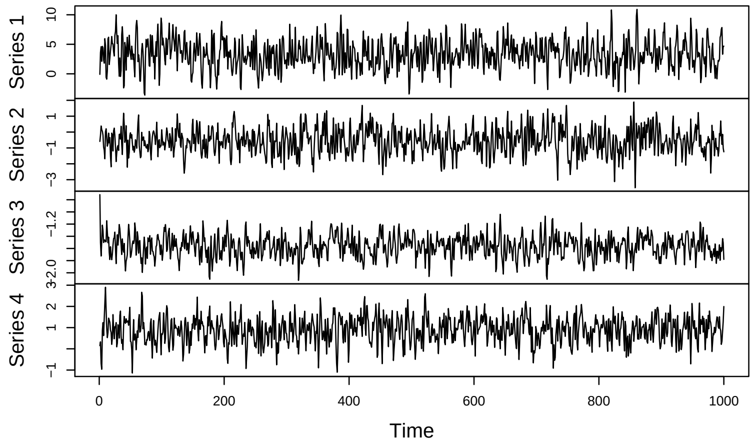
MA(1): μ = 0, σ = 1



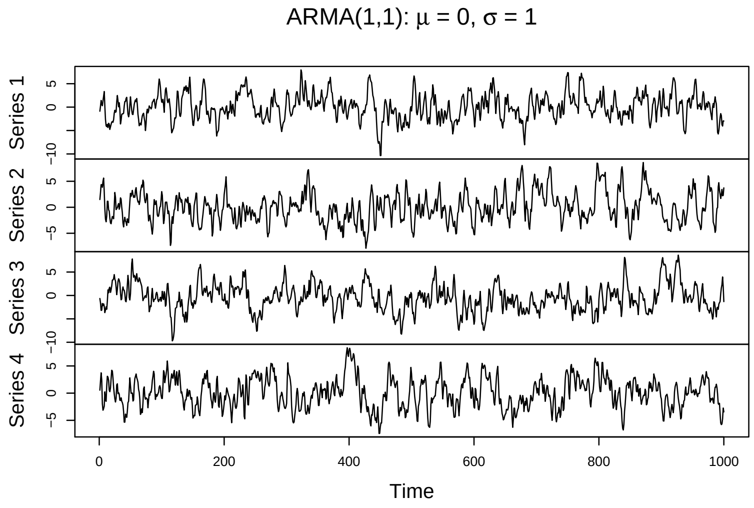
MA(1): μ = ϕ(W), σ = 1

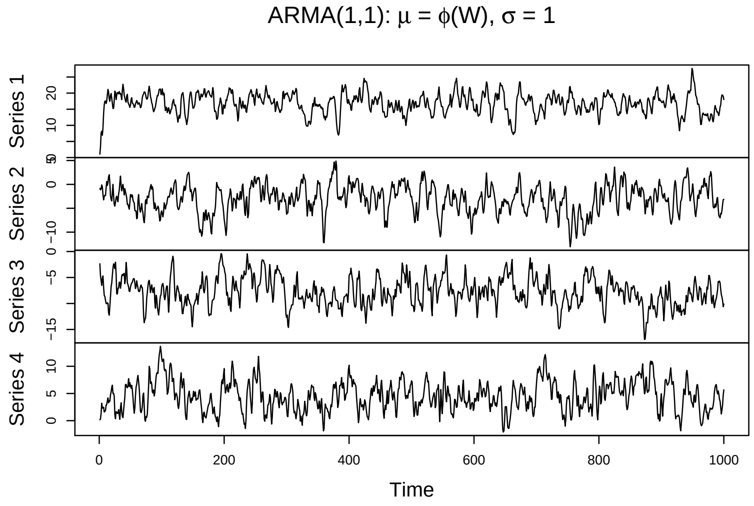


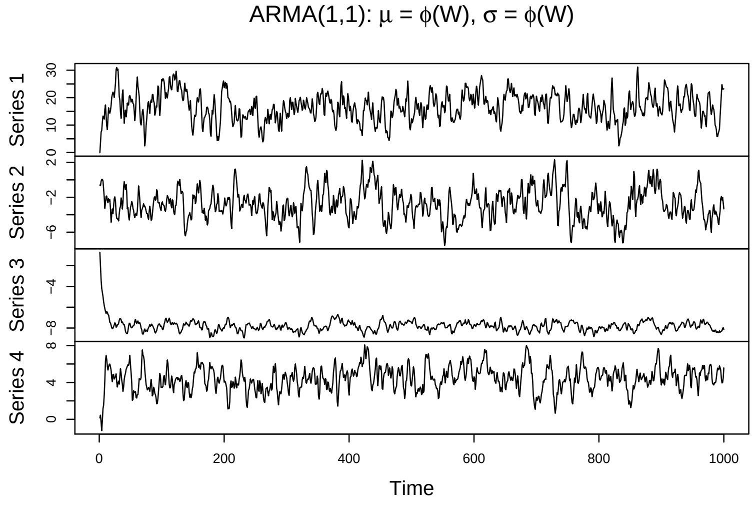
MA(1): μ = ϕ(W), σ = ϕ(W)



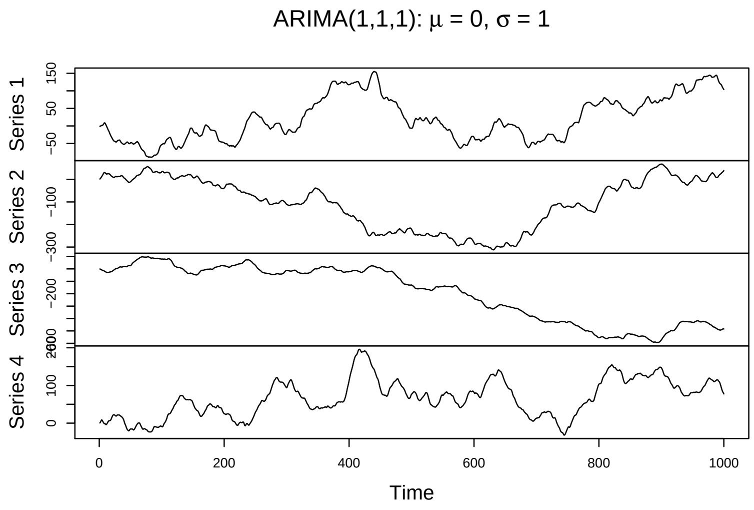
3. An autoregressive moving average model of order (1, 1), combining the two above.


ARMA(1,1): μ = 0, σ = 1

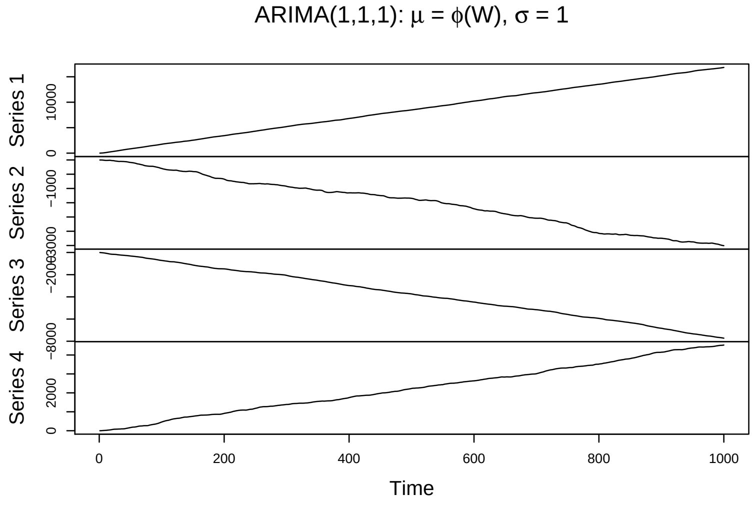

ARMA(1,1): μ = ϕ(W), σ = 1


ARMA(1,1): μ = ϕ(W), σ = ϕ(W)

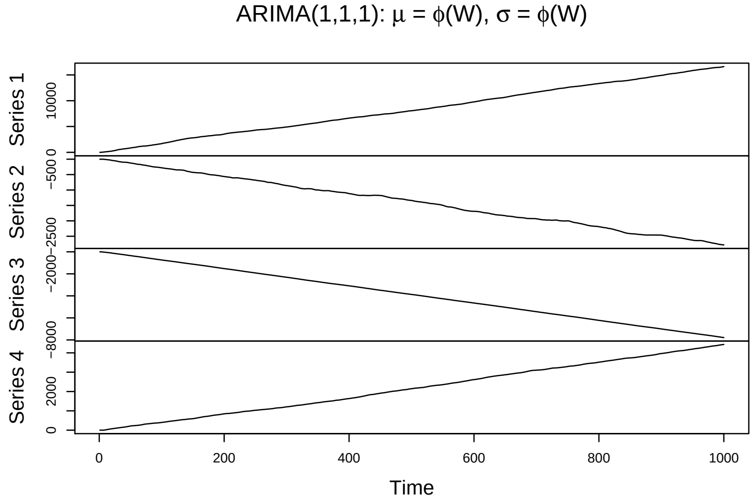
4. An ARIMA(1, 1, 1) model that is the cumulative sum of the ARMA(1, 1), so the first difference of the time series is stationary.



ARIMA(1,1,1): $\mu = 0$, $\sigma = 1$



ARIMA(1,1,1): $\mu = \phi(W)$, $\sigma = 1$



ARIMA(1,1,1): $\mu = \phi(W)$, $\sigma = \phi(W)$

Simulations are based on those presented in free range statistics

MIMIC Simulation

We consider the following outcomes of interest: `abpmean`, `abpsys`, and `abpdias`. For each outcome of interest, we generate a table where each row corresponds to a subject and the columns are the model parameters.

Also, for each outcome of interest, we generate another table where each row corresponds to a subject and the columns are measures of accuracy of the model fit on the training and test data. The first 80% of data is split into a training set, and last 20% into a test set. The training set performance is the in sample performance (i.e. it is computing performance using the data that it was fit with) and we can compare this performance to the test set. The accuracy measures pertain to predicting the test data and we consider two mechanisms for this prediction:

1. multi-step – forecasts are for as many time points in the test set. This procedure applies the already fitted model and repeatedly feeds the predicted data point back into the prediction equation to get the next prediction.
2. one-step – forecasts are for as many time points in the test set. This procedure applies the already fitted model and predicts "one step ahead" forecasts by predicting the next outcome and then adds the actual outcome to the prediction equation for next outcome prediction.

Note both of these procedures update the prediction equation with the new data point, and do not changing the model coefficients. You can do a full refit of the model each time you get a new data point and then predict with that, but I do not consider that procedure in this assessment of model accuracy.

The measures of accuracy are the mean error (`ME`), root mean squared error (`RMSE`), mean absolute error (`MAE`), mean percentage error (`MPE`), mean absolute percentage error (`MAPE`), mean absolute scaled error (`MASE`) and the first-order autocorrelation coefficient (`ACF1`). More here on evaluating forecast accuracy.

Later on, we consider a more sophisticated version of training/test sets with time series cross-validation, which incorporates a series of test sets, each consisting of a single observation. We could choose a forecasting model for simulation by identifying the model with the smallest RMSE based on time series cross-validation.

ARIMA

We employ the `auto.arima` function to fit the "best" ARIMA model to a univariate time series (i.e., each subject in the MIMIC data). In this case, the best model is defined as the one that has the least information loss relative to the true model. Information criteria (IC) are estimates of the Kullback Leibler information loss. The best known IC is the Akaike IC $AIC = 2ln(l) + 2k$ and its corrected form for small sample sizes, $AICc = AIC + \frac{2k(k+1)}{Nk1}$, as well as its Bayesian alternative, $BIC = 2ln(l) + ln(N)k$, where l is the maximum likelihood of the model, k is the number of degrees of freedom (or independently adjusted parameters) in the model and N is the number of observations. Is there any reason to prefer the AIC or BIC over the other?

`auto.arima` selects the optimal autoregressive and moving average orders p and q based on a chosen information criterion (AICc by default) from a local search over a few regions of values.

```r
library(here)
library(forecast)
load(here::here("Data", "mimic.Rdata"))
set.seed(4197)

# sample n individuals with t hours of data
dat <- sample_n_t(mimic, n = 500, t = 5)

arima_abpsys <- run_auto_arima(df = dat, outcome = "abpsys")
arima_abpdias <- run_auto_arima(df = dat, outcome = "abpdias")
arima_abpmean <- run_auto_arima(df = dat, outcome = "abpmean")
```

Table 1: ARIMA Coefficients with Systolic BP Outcome

	10013	10241	10315	10320	10342	10384	10419	10423
ma1	-0.4292	0.2173	0.5862	-0.9831	-0.1658	-0.0797	-0.9584	-0.9811
ma2	-0.3695	NA	-0.5507	NA	-0.8071	-0.0151	NA	NA
ar1	NA	0.4261	-1.2101	0.4076	0.1545	-0.3064	0.3602	0.7616
intercept	NA	106.4415	NA	NA	NA	NA	NA	NA
ar2	NA	NA	-0.5725	0.2777	0.1513	-0.3631	0.1717	NA
ma3	NA	NA	-0.6887	NA	NA	-0.7377	NA	NA
ar3	NA	NA	NA	NA	0.1459	NA	NA	NA
ma4	NA	NA	NA	NA	NA	NA	NA	NA
ma5	NA	NA	NA	NA	NA	NA	NA	NA
ar4	NA	NA	NA	NA	NA	NA	NA	NA
ar5	NA	NA	NA	NA	NA	NA	NA	NA
drift	NA	NA	NA	NA	NA	NA	NA	NA

Table 2: ARIMA Accuracy with Systolic BP Outcome

	10013	10241	10315	10320	10342	10384	10419	10423
ME_test	-27.3191	-2.2745	-0.7373	17.8977	-3.1209	7.9857	-3.3792	35.1391
RMSE_test	32.7164	20.7865	11.9294	19.3771	16.7804	11.4891	19.5924	50.7121
MAE_test	27.3191	9.5712	9.9015	17.9457	8.2885	9.8881	12.3942	38.4045
MPE_test	-Inf	-Inf	-1.9846	13.3379	-Inf	6.8510	-Inf	-Inf
MAPE_test	Inf	Inf	9.5639	13.3820	Inf	8.9596	Inf	Inf
MASE_test	6.0272	1.2675	0.7534	2.3114	1.1700	2.2274	1.1867	5.5320
ME_onestep	1.2767	-1.0320	-1.6298	2.8992	-1.7189	0.4557	-4.1050	-1.7959
RMSE_onestep	19.0352	25.5694	9.1009	4.4097	36.9873	6.8448	18.4357	21.1504
MAE_onestep	8.7445	9.7700	6.9636	3.4248	17.8504	5.6742	8.1000	10.9722
MPE_onestep	-Inf	-Inf	-2.2894	2.1687	-Inf	0.0182	-Inf	-Inf
MAPE_onestep	Inf	Inf	6.7731	2.6040	Inf	5.4156	Inf	Inf
MASE_onestep	1.1129	0.9344	2.4026	1.2878	2.3347	1.2753	1.1333	1.2816
ACF1_onestep	-0.4287	-0.5274	0.9021	0.2698	-0.8977	0.0737	-0.3102	-0.2009
ME_train	0.3861	-0.0296	1.7428	1.6700	-0.3075	0.8264	0.3937	-0.6116
RMSE_train	13.4179	18.6723	25.3693	17.5788	16.8892	11.9183	26.6353	17.6224
MAE_train	6.0286	8.6114	15.1259	8.5608	8.4392	5.6974	12.5220	8.8189
MPE_train	-Inf	-Inf	-Inf	-Inf	NA	-Inf	-Inf	-Inf
MAPE_train	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
MASE_train	1.3301	1.1404	1.1509	1.1026	1.1913	1.2834	1.1989	1.2703
ACF1_train	0.0203	0.0008	0.0055	-0.0182	-0.0122	0.0063	-0.0069	-0.0009

Table 3: ARIMA Coefficients with Diastolic BP Outcome

	10013	10241	10315	10320	10342	10384	10419	10423
ME_test	4.3131	-2.9033	1.4731	6.8951	-5.3764	0.5745	-2.2236	19.3176
RMSE_test	19.3365	9.1553	5.8805	7.3064	10.0911	3.4796	8.8107	25.9581
MAE_test	14.0036	4.1357	5.0330	6.9920	6.2204	2.8297	5.1889	21.0724
MPE_test	-Inf	-Inf	1.4634	10.0636	-Inf	0.6085	-Inf	-Inf
MAPE_test	Inf	Inf	7.6987	10.2296	Inf	4.5221	Inf	Inf
MASE_test	8.7339	1.3518	0.6383	1.6182	1.7741	0.9645	1.2145	5.4300
ME_onestep	-0.8302	-1.4453	-0.6418	0.8778	-0.6487	-0.2866	-2.2406	-1.0322
RMSE_onestep	15.5020	10.6726	4.4281	1.9059	18.5791	3.9279	8.0818	11.3555
MAE_onestep	7.6284	4.1975	3.2082	1.4806	8.6283	3.1243	3.2753	5.9010
MPE_onestep	-Inf	-Inf	-1.4628	1.2431	-Inf	-0.7147	-Inf	-Inf
MAPE_onestep	Inf	Inf	5.0787	2.1846	Inf	5.0615	Inf	Inf
MASE_onestep	1.2141	1.0584	2.0353	1.1128	2.2777	1.1939	1.2108	1.3702
ACF1_onestep	-0.0586	-0.5298	0.8654	0.1806	-0.8895	-0.2553	-0.2526	-0.1911
ME_train	0.2208	0.0149	0.9957	0.8532	0.1653	0.0122	0.7671	-0.4345
RMSE_train	4.8366	8.1490	15.8748	9.7261	9.0713	7.1036	11.3424	10.0018
MAE_train	2.0653	3.5668	9.3688	5.0080	4.4931	3.4955	5.1300	4.9348
MPE_train	-Inf	-Inf	-Inf	-Inf	NA	-Inf	-Inf	-Inf
MAPE_train	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
MASE_train	1.2881	1.1658	1.1883	1.1590	1.2815	1.1914	1.2007	1.2716
ACF1_train	0.0032	-0.0007	0.0052	-0.0149	-0.0032	0.0099	-0.0082	0.0095

Table 4: ARIMA Accuracy with Diastolic BP Outcome

	10013	10241	10315	10320	10342	10384	10419	10423
ME_test	4.3131	-2.9033	1.4731	6.8951	-5.3764	0.5745	-2.2236	19.3176
RMSE_test	19.3365	9.1553	5.8805	7.3064	10.0911	3.4796	8.8107	25.9581
MAE_test	14.0036	4.1357	5.0330	6.9920	6.2204	2.8297	5.1889	21.0724
MPE_test	-Inf	-Inf	1.4634	10.0636	-Inf	0.6085	-Inf	-Inf
MAPE_test	Inf	Inf	7.6987	10.2296	Inf	4.5221	Inf	Inf
MASE_test	8.7339	1.3518	0.6383	1.6182	1.7741	0.9645	1.2145	5.4300
ME_onestep	-0.8302	-1.4453	-0.6418	0.8778	-0.6487	-0.2866	-2.2406	-1.0322
RMSE_onestep	15.5020	10.6726	4.4281	1.9059	18.5791	3.9279	8.0818	11.3555
MAE_onestep	7.6284	4.1975	3.2082	1.4806	8.6283	3.1243	3.2753	5.9010
MPE_onestep	-Inf	-Inf	-1.4628	1.2431	-Inf	-0.7147	-Inf	-Inf
MAPE_onestep	Inf	Inf	5.0787	2.1846	Inf	5.0615	Inf	Inf
MASE_onestep	1.2141	1.0584	2.0353	1.1128	2.2777	1.1939	1.2108	1.3702
ACF1_onestep	-0.0586	-0.5298	0.8654	0.1806	-0.8895	-0.2553	-0.2526	-0.1911
ME_train	0.2208	0.0149	0.9957	0.8532	0.1653	0.0122	0.7671	-0.4345
RMSE_train	4.8366	8.1490	15.8748	9.7261	9.0713	7.1036	11.3424	10.0018
MAE_train	2.0653	3.5668	9.3688	5.0080	4.4931	3.4955	5.1300	4.9348
MPE_train	-Inf	-Inf	-Inf	-Inf	NA	-Inf	-Inf	-Inf
MAPE_train	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
MASE_train	1.2881	1.1658	1.1883	1.1590	1.2815	1.1914	1.2007	1.2716
ACF1_train	0.0032	-0.0007	0.0052	-0.0149	-0.0032	0.0099	-0.0082	0.0095

Table 5: ARIMA Coefficients with Mean BP Outcome

	10013	10241	10315	10320	10342	10384	10419	10423
ma1	-0.8963	NA	-0.3206	NA	-0.2312	0.6246	-0.4628	-0.7885
ma2	-0.0857	NA	-0.3589	NA	-0.6135	NA	0.0970	-0.5976
ma3	-0.0195	NA	-0.0325	NA	NA	NA	0.1337	0.3971
ma4	0.2101	NA	0.2128	NA	NA	NA	-0.2892	NA
ar1	NA	0.4865	NA	0.5024	NA	NA	NA	0.8332
ar2	NA	0.1858	NA	NA	NA	NA	NA	NA
ar3	NA	-0.1504	NA	NA	NA	NA	NA	NA
intercept	NA	70.0392	NA	80.1328	NA	71.1340	NA	NA
ma5	NA	NA	-0.2811	NA	NA	NA	0.0980	NA
ar4	NA	NA	NA	NA	NA	NA	NA	NA
drift	NA	NA	NA	NA	NA	NA	NA	NA
ar5	NA	NA	NA	NA	NA	NA	NA	NA

Table 6: ARIMA Accuracy with Mean BP Outcome

	10013	10241	10315	10320	10342	10384	10419	10423
ME_test	-1.3269	-5.3051	1.3337	10.8283	-8.6472	7.8061	1.1651	20.0119
RMSE_test	11.6017	6.8034	7.7586	11.6376	11.2712	9.3070	6.8437	30.5179
MAE_test	9.4188	5.7027	6.5544	10.9735	9.9480	8.1129	5.3203	22.4899
MPE_test	-4.6561	-8.6519	0.7203	11.7407	-13.5452	9.4996	0.8158	16.5390
MAPE_test	13.9842	9.1890	8.2947	11.9343	14.8645	9.9505	6.9910	20.0504
MASE_test	2.5413	1.3787	1.5798	2.1877	2.4415	1.8263	2.1482	5.4966
ME_onestep	-0.4602	-2.5945	-0.5821	5.3122	-0.9342	4.8850	-0.5632	-0.8105
RMSE_onestep	12.3624	3.6413	3.7860	6.1275	5.0852	6.3878	3.7344	13.1873
MAE_onestep	7.7336	3.1171	2.8010	5.6176	3.0139	5.5119	2.1430	8.5269
MPE_onestep	-2.5802	-4.2485	-1.0334	5.7059	-1.8892	5.9095	-0.8999	-2.2572
MAPE_onestep	12.2709	4.9727	3.6317	6.1089	4.3149	6.8043	2.8420	8.9691
MASE_onestep	2.8952	2.2319	1.4029	2.8499	1.2072	1.6474	1.1751	1.6996
ACF1_onestep	0.8890	0.4926	0.6510	0.3408	0.2189	0.1184	0.1118	-0.4305
ME_train	-0.2661	0.0031	0.4092	0.0362	0.2003	0.0287	0.0399	-0.5363
RMSE_train	16.8447	9.7105	7.5800	13.0367	8.9977	14.9779	4.7079	10.0404
MAE_train	5.3915	4.4321	4.2733	5.4967	4.9848	3.9282	2.7053	5.1852
MPE_train	-1.8448	-1.2396	-0.0522	-1.6043	-0.7942	-3.9160	-0.1238	-1.6770
MAPE_train	7.1047	5.8065	5.1361	6.5144	6.3510	7.4707	3.2370	6.2042
MASE_train	1.4547	1.0715	1.0300	1.0959	1.2234	0.8843	1.0924	1.2673
ACF1_train	0.0504	0.0035	0.0128	-0.0306	0.0554	-0.0138	0.0028	0.0101

Time series cross-validation

Since we are interested in models that produce good 30-minute-ahead forecasts, we can employ a cross-validation procedure based on a rolling forecasting origin which allows multi-step errors. More details here.

Moving forward

- MIMIC data without the time gap
- Defining a hypotensive episode