

# 2D-Interval Predictions for Time Series\*

Luis Torgo  
LIAAD/INESC Porto LA - FC, University of Porto  
Rua Campo Alegre, 1021/1055  
4169-007 Porto, Portugal  
ltorgo@dcc.fc.up.pt

Orlando Ohashi<sup>†</sup>  
LIAAD/INESC Porto LA  
Rua Dr. Roberto Frias, 378  
4200-465 Porto, Portugal  
ohashijr@dcc.fc.up.pt

## ABSTRACT

Research on time series forecasting is mostly focused on point predictions - models are obtained to estimate the expected value of the target variable for a certain point in future. However, for several relevant applications this type of forecasts has limited utility (e.g. costumer wallet value estimation, wind and electricity power production, control of water quality, etc.). For these domains it is frequently more important to be able to forecast a range of plausible future values of the target variable. A typical example is wind power production, where it is of high relevance to predict the future wind variability in order to ensure that supply and demand are balanced. This type of predictions will allow timely actions to be taken in order to cope with the expected values of the target variable on a certain future time horizon. In this paper we study this type of predictions - the prediction of a range of expected values for a future time interval. We describe some possible approaches to this task and propose an alternative procedure that our extensive experiments on both artificial and real world domains show to have clear advantages.

## Categories and Subject Descriptors

H.4 [Information Systems Applications]: Miscellaneous

## General Terms

Algorithms

## Keywords

Time Series, Forecasting, Prediction

---

\*This work is partially supported by FEDER funds through COMPETE program and by FCT national funds in the context of the project PTDC/EIA/68322/2006.

<sup>†</sup>The work of Orlando Ohashi is supported by the PhD grant SFRH/BD/61795/2009 from FCT.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

KDD'11, August 21–24, 2011, San Diego, California, USA.  
Copyright 2011 ACM 978-1-4503-0813-7/11/08 ...\$10.00.

## 1. INTRODUCTION

Time series forecasting is a key modeling technique in many real world applications. One of the goals of developing time series models is to obtain predictions of the future values of the series. The most common form of predictions are the so-called point predictions - at time  $t$  try to forecast the value of the series for time  $t + h$ , where  $h$  is the forecasting horizon. However, some applications require a different type of forecasts. Namely, users may be interested in obtaining an interval of plausible values for time  $t + h$ . Moreover, other applications may even require this type of interval of values for a certain future interval of time  $[t + h, t + h + k]$ . We will call this latter type of predictions 2D-interval predictions where the goal is to forecast the expected interval of values of the time series for a certain future time interval. How to handle these 2D-interval predictions is the subject of this paper. Figure 1 briefly describes the differences between these three types of forecasts.

Existing work on time series forecasting essentially focus on (i) point predictions (e.g. [6]), (ii) on predicting an interval where the future value is expected to be with a certain probability (e.g. [5]), or even (iii) on generalizing the latter to density forecasts [16]. To the best of our knowledge there is no established method for forecasting an interval of values for an interval of time. Addressing this limitation is the main goal of this paper. The lack of a method to handle these problems is rather surprising given the amount of relevant applications that could benefit from this type of predictions. For instance, any application requiring some form of production planning for a certain demand scenario, will find this type of predictions of high utility. This includes areas like manufacturing in general, energy production, water distribution, etc. For example, in wind power production it is important to predict the future wind variability in order to ensure that supply and demand are balanced [3, 18]. In power production it is common to place bids for the future production. In this context, the forecast of the future wind variability must be accurate not to receive a penalty from any deviation between production and demand. To make optimal decisions, a model that is able to predict the quantiles of the distribution gives much more information than single point predictors [3]. Similar problems are faced in electricity markets, where it is more important to predict the interval of demand than a single value [17]. In inventory management, over-production may lead to inventory costs while under-production may originate unsatisfied demand and lost profits [4]. Other relevant application areas include customer wallet estimation [14] or computer network

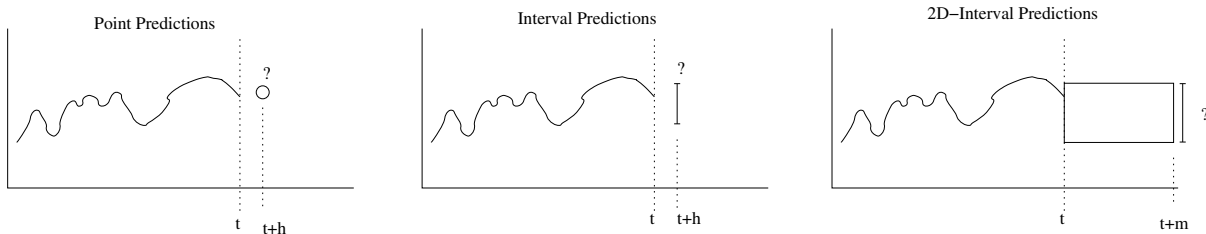


Figure 1: Three types of time series prediction tasks.

traffic analysis [21]. Several investment-related applications (e.g. financial markets) will also find this type of forecasts of great use.

From a theoretical perspective we are talking about forecasting the distribution of the values of the target time series variable for a future time window. The key difference from existing work on density forecasting (e.g. [16]) is the fact that we want to estimate a density for a time interval and not for a single point in time. The applicability to real world scenarios motivates this difference. In this work we restrict this general scenario to the prediction of some descriptive statistics of this distribution. Namely, we will focus on forecasting a kind of interval of “normality” of the values of the series for a certain future time interval. We will represent this interval of normality by the 1st and 3rd quartiles of the variable distribution, though our proposal could be applied to any quantile.

The main contributions of our work are: (1) to increase the awareness of the data mining community to a high-impact task that was not addressed before; and (2) to propose a general method for addressing this class of time series forecasting problems. In the context of the presentation of our proposal we describe an extensive set of experiments that we have carried out to demonstrate the validity of our approach under a wide range of scenarios.

## 2. PREDICTING 2D-INTERVALS

A 2D-interval is a range where the values of the time series are expected to be included in a future time window with high probability (c.f. Figure 1). The goal of this concept is to provide the user with a plausible range of values of the time series in this future period. It is thus a form of summary statistic of the unknown future distribution of the values of the series for the target forecasting horizon. We will address this general problem by simplifying it into predicting the 1st and 3rd quartiles of this unknown distribution. These two statistics provide us with an interval where a significant number of the values should be contained (assuming a near-normal distribution of the values of the target variable).

### 2.1 Possible Approaches to the Problem

As mentioned before, we are not aware of any existing approach to the 2D-interval prediction problem. Nevertheless, we can try to apply the existing time series forecasting techniques to obtain this type of forecasts.

Point estimates can be iterated to obtain predictions for more than one point in the future (known as iterated predictions e.g. [6]). At time  $t$  we obtain a prediction for time  $t + 1$ . This prediction can be incorporated into the training data available to the model as if it was true, and a new step ahead prediction can be obtained, which in effect would

correspond to a prediction for time  $t + 2$ . These iterations can continue until we get a point prediction for all the forecasting interval. Using these predictions we can obtain the statistics we want to have a 2D-interval forecast. This is a simple approach that can be applied to any existing time series modeling technique. The main drawback of this strategy is that by incorporating the one-step ahead predictions into the training data we are potentially amplifying the prediction errors of the models.

Another plausible approach to our problem is to decompose it into several prediction tasks. If we want a 2D-interval for a time window of  $k$  steps in the future, then we can transform this into  $k$  prediction tasks with each being addressed by a different model designed to obtain a point prediction for  $t + i$ , where  $i \in [1, k]$ . Given these  $k$  models, to obtain a 2D-interval forecast at time  $t$  we use the  $k$  models predictions and calculate the necessary statistics to obtain the interval. The main drawback of this approach is the computational cost of obtaining  $k$  different models. This may be a critical issue with high-frequency data as it is often the case in time series problems.

Finally, we will also consider a naive approach as a kind of baseline method. A 2D-interval forecast at time  $t$  for the interval  $[t + 1, t + k]$  can be obtained by using the statistics calculated with the known values of the series in the interval  $[t - k, t]$ . This can be regarded as a kind of random walk approach to our problem and we will expect other alternatives to clearly outperform this baseline method.

### 2.2 Our Proposal

In order to obtain 2D-interval predictions we propose to directly forecast some summary statistics of the distribution of the time series values for the target future time window. Let  $Q_\alpha^k$  and  $Q_\beta^k$  be the  $\alpha$  and  $\beta$  unknown quantiles of the time series values for a future time window of length  $k$ , respectively. These two values establish an interval where  $|\beta - \alpha| \times 100\%$  of the series values are supposed to be included. Assuming a near normal distribution of the values of the target variable the interval between the 1st and 3rd quartiles ( $Q_{0.25}$  and  $Q_{0.75}$ ) will contain roughly 50% of the values of the series. We will use the predicted values for these two distribution statistics as the source for obtaining 2D-interval predictions. This means that we will directly forecast these quantile statistics instead of calculating them from predictions of the target time series. Our proposal is thus to define two prediction problems  $Q_\alpha^k = f(v_1, \dots, v_a)$  and  $Q_\beta^k = f(v_1, \dots, v_a)$ , where  $v_1, \dots, v_a$  are a set of descriptor variables, and  $Q_\alpha^k$  ( $Q_\beta^k$ ) is the  $\alpha$  ( $\beta$ ) quantile of the time series variable for the next  $k$  time points, i.e. the estimated quantile for the time interval  $[t, t + k]$ . In order to be able to obtain models to forecast these quantiles we

need to have a set of training data where these values are known. Thus for “preparing” the training sample at time  $t$  we will need information on both the selected predictor variables but also on the target time series values in the period  $[t + 1, t + k]$ , where  $k$  is the time length of the target 2D-intervals.

The methods presented in Section 2.1 are all based on predicting the value of the target time series for a future time interval. This is done either by iterated predictions or by obtaining several different models. Either way it is based on these predictions that we calculate the summary statistics to obtain the 2D-intervals. In our approach we directly predict these statistics. We claim that this is an easier prediction task as quantile statistics have a distribution that is smoother than the original variables from which they are calculated. In effect, quantiles are known to be robust to a few extreme values, thus smoothing-out these variations on the original time series. This means that the distribution of the quantile variables that we use as targets is clearly more “well behaved” than the distribution of the original series. We expect this to bring an advantage to our proposal, because it should simplify the task of obtaining 2D-intervals. The experiments described in Section 3 are designed to test this hypothesis.

The choice of the predictor variables  $v_1, \dots, v_a$  is not part of our proposal. They should be selected with the goal of trying to optimize the performance of the selected modeling tool in the task of forecasting the target variable. This decision is not different from what is necessary on any other time series prediction task. Our proposal only changes the target variable. In our experiments we will include as predictors recent past values of the time series and also past values of some distribution statistics as we will see in Section 3.

### 3. EXPERIMENTAL EVALUATION

In this section we evaluate our proposed method for obtaining 2D-interval predictions under a large set of experimental setups. The main goal of these experiments is to check the validity of the hypothesis that directly predicting the quantile statistics is a better method of obtaining intervals of variation of a time series for a future time window. All code, data and extra results not fitting here are provided in a web page<sup>1</sup> associated with this paper, to ensure that our work is replicable.

#### 3.1 Experimental methodology

##### 3.1.1 Models

We have tried to select a wide range of modeling approaches to test our method. The idea is to confirm its validity independently of the technique used to forecast the  $Q_p^k$  quantiles. All the used tools are freely available in the R software environment [15], which ensures easy replication of our work. The following is a list of the methods used in our experiments:

**Random Walk (RW)** - a simple baseline method that uses the quantiles estimated with the last  $k$  time series values as predictions for the quantiles of the next  $k$  time points;

**Regression Trees (RT)** - a regression tree (e.g. [1]) based on the R package `rpart` [19]. In our experiments we have used an interface to the `rpart` function provided in package `DMwR` [20] and have tried 4 different variants by using the parameter `se` that controls the level of pruning with values 0, 0.5, 1 and 1.5.

**Support Vector Machines (SVM)** - an implementation of SVMs (e.g. [7]) available in the R package `e1071` [8]. We have tried 20 variants by using the parameter `cost` with the values 1, 5, 10, 50, 100 and the parameter `gamma` with the values 0.001, 0.01, 0.05 and 0.1.

**Random Forest (RF)** - an implementation of random forests [2] available in the R package `randomForest` [11]. We have used 3 variants of the parameter `ntree` with the values 500, 1000 and 1500.

**Quantile Regression Forests (QRF)** - a random forest variant [12] designed to optimize the prediction of quantiles. We have used the implementation of these models available in the R package `quantregForest` [13]. We have tried 3 variants of the parameter `ntree` with the values 500, 1000 and 1500.

##### 3.1.2 Evaluation Metrics

There is an extensive literature on evaluation metrics for single point prediction models. Most measures compare the true and predicted values and eventually contrast the performance of the model being evaluated against some baseline. Our prediction task is different as we have mentioned before. We are addressing the prediction of a 2D-interval by means of the 1st and 3rd quartiles, which means there are some similarities with the goals of quantile regression [10]. However, in quantile regression the goal is to obtain point predictions of the quantiles. The evaluation of quantile regression models is usually carried out with the help of Equation 1. It can be easily shown that the value of  $L_\alpha(y, \hat{y})$  is optimized by predicting the quantile  $Q_\alpha$  (i.e.  $\hat{y} = Q_\alpha$ ).

$$L_\alpha(y, \hat{y}) = \begin{cases} \alpha \cdot (y - \hat{y}) & \text{if } y \geq \hat{y} \\ (1 - \alpha) \cdot (\hat{y} - y) & \text{otherwise} \end{cases} \quad (1)$$

In this context, if we want to estimate the values of  $Q_{0.25}$  and  $Q_{0.75}$  for a certain period of time  $k$  we can evaluate the predictions of our models ( $\hat{Q}_{0.25}^k$  and  $\hat{Q}_{0.75}^k$ ) using Equation 1, given the true target variable values  $y_{t+1}, \dots, y_{t+k}$ . Moreover, if we are given a test set we can calculate the total quantile error (TQE) of our 2D-interval predictions as follows,

$$TQE = \sum_{i=1}^n \left[ \sum_{j=i}^{i+k} L_{0.25}(y_j, \hat{Q}_{0.25,i}^k) + \sum_{j=i}^{i+k} L_{0.75}(y_j, \hat{Q}_{0.75,i}^k) \right] \quad (2)$$

where  $\hat{Q}_{\alpha,t}^k$  is the  $\alpha$  quantile prediction for the future  $k$ -length interval starting at time  $t$ .

We have also compared our alternative models using the mean absolute quantile (MAQ) deviation of the model predictions, i.e.

$$MAQ = \frac{1}{2n} \left[ \sum_{i=1}^n |Q_{0.25,i}^k - \hat{Q}_{0.25,i}^k| + |Q_{0.75,i}^k - \hat{Q}_{0.75,i}^k| \right] \quad (3)$$

<sup>1</sup><http://www.dcc.fc.up.pt/~ltorgo/KDD11>

**Table 1: Benefit matrix.**

	<i>low</i>	<i>normal</i>	<i>high</i>
<i>low</i>	2	-1	-2
<i>normal</i>	-1	1	-1
<i>high</i>	-2	-1	2

This evaluation metric can be easily obtained by calculating the observed 2D-interval quantiles and comparing these with the predictions of our models. This statistic will measure the average absolute error of our quantile predictions when compared to the true observed quantiles.

The real world applications that we target with this proposal of 2D-interval predictions have some particularities that are not completely captured by the statistics of Equations 2 and 3. Namely, there are usually costs and benefits associated with the predictions of the models. In effect, the predicted intervals are frequently used to carry out some actions (e.g. production planning) that may result in costs or benefits depending on the accuracy of the predictions. We will also evaluate the 2D-interval predictions taking these factors into account. The predicted intervals divide the values in three classes: unusually high or low values, and normal values that are within the intervals. This means that given the predictions  $\hat{Q}_{0.25}^k$  and  $\hat{Q}_{0.75}^k$  we can discretize the series values into these three classes. Accordingly, we can look at the observed values in the  $k$ -length interval and calculate the real  $Q_{0.25}^k$  and  $Q_{0.75}^k$  values. Using these values we can calculate the true class labels of each value in the  $k$  period. Elkan [9] has established the theoretical grounds for cost-sensitive learning. The proposed framework is based on the concept of benefit matrix. This matrix sets the benefits of all accurate predictions, as well as the costs (negative benefits) of the errors. We will also use this setup to evaluate the 2D-interval predictions of our models. We discretize the continuous values of the target time series using the method described above and then calculate the *Utility* of the predictions as the total sum of benefits using the matrix in Table 1.

### 3.2 Experiments with Artificial Data

The first set of experiments that we will describe involves the use of artificially created data sets. The goal was to test our hypothesis on a diverse range of time series with different dynamic regimes. Figure 2 shows the seven artificial time series we have generated and used in our experiments, each with 3000 values<sup>2</sup>. The gray lines on each graph try to describe the type of regimes in terms of trend and variability that have guided the generation process. As you can observe these series have rather different dynamic regimes in terms of these two important properties.

We have applied the models described in Section 3.1.1 to these 7 problems, using the 3 alternative methods for obtaining 2D-intervals. Our goal is to check which is the best method for obtaining this type of predictions: (i) iterating the model over the  $k$  window; (ii) obtaining  $k$  different models; or (iii) our proposed method of predicting directly the distribution statistics defining the intervals. For each experimental setup, we have estimated the values of the 3 evaluation metrics (*TQE*, *MAQ* and *Utility*) described in

<sup>2</sup>The R code used to generate the data sets as well as the data sets themselves, are available at the paper web page.

Section 3.1.2 using a Monte Carlo simulation. Namely, we have randomly selected 10 time points within each data set. For each of these 10 points we have used the previous 365 values of the series to obtain the models that were then used to make 2D-interval predictions for the next 90 points using a sliding window approach. At each of these 90 prediction points the goal was to estimate the 2D-interval for a future time window of 10, 20 and 30 time points. The results we present for these 3 different values of  $k$  (the time length of the 2D-intervals) are averages over these test sets with 90 points, on the 10 Monte Carlo repetitions (i.e. at randomly selected points of the full time series).

The goal for all alternatives is to obtain an estimate of the 1st and 3rd quartiles for the target time intervals ( $k = 10, 20$  and  $30$ ). However, depending on the approach followed, these estimates are obtained using a different method as explained in Section 2. Still, all alternatives will use the same predictor variables that were selected with the goal of trying to provide useful information on the recent dynamics of the time series and also past values of  $k$ -length descriptive statistics. The used prediction tasks are described by the following general equation,

$$TGT = f(Y_{t-1}, \dots, Y_{t-10}, Q_{0.25,t}^{-k}, Q_{0.75,t}^{-k}, \bar{Y}^{-k}, \sigma_Y^{-k}) \quad (4)$$

where  $Q_{\alpha,t}^{-k}$  is the value of the  $\alpha$  quantile calculated using the past  $k$  values of the series at time  $t$ ,  $\bar{Y}^{-k}$  is the average time series value on the same past window,  $\sigma_Y^{-k}$  the respective standard deviation, and  $Y_{t-1}, \dots, Y_{t-10}$  are the last 10 values of the series. The target variable (*TGT*) will be different depending on the approaches. For the iterated predictions this will be the next value of the series,  $Y_{t+1}$ . For the  $k$  models the targets will be  $Y_{t+i}$  where  $i \in [1, k]$  for each of the  $k$  models. For our proposal there will be two models, one predicting the 1st quartile on the 2D-interval,  $Q_{0.25,t}^k$ , and the other the 3rd quartile,  $Q_{0.75,t}^k$ .

All model variants were given exactly the same data in the context of the Monte Carlo simulation, with the exception of the target variable. Given the large set of experimental setups (3 metrics, 7 problems, 3 values for  $k$ ) we can not afford to present all results given the space restrictions applicable. Nevertheless, we can confirm that in all setups we have observed a similar pattern of results/conclusions. The full experimental results can be checked at the Web page associated with the paper.

Figure 3 shows the results for 3 different setups. We have selected one graph for each series, metric and interval length. The graph on the left shows the *TQE* scores of all model variants on the first time series when predicting a 2D-interval of length 10. The results of the models are grouped in three batches, one for each different approach we are comparing: the iterated approach (“iterated” on the graphs), the use of several models (“k-models” on the graphs), and our proposal of directly predicting the quantiles (“quantiles” on the graphs). For each of these batches we show the results of all model variants described before. A vertical dashed line marks the best score on each graph and the result of the baseline *RW* model is given as a sub-title of the graphs. The middle graph shows the same type of results this time for the *MAQ* metric, 4th time series and  $k = 20$ , while the graph on the right presents the results in terms of *Utility* on the 7th series for  $k = 30$ . Note that while for the first two metrics lower scores are better, for utility it is the opposite.

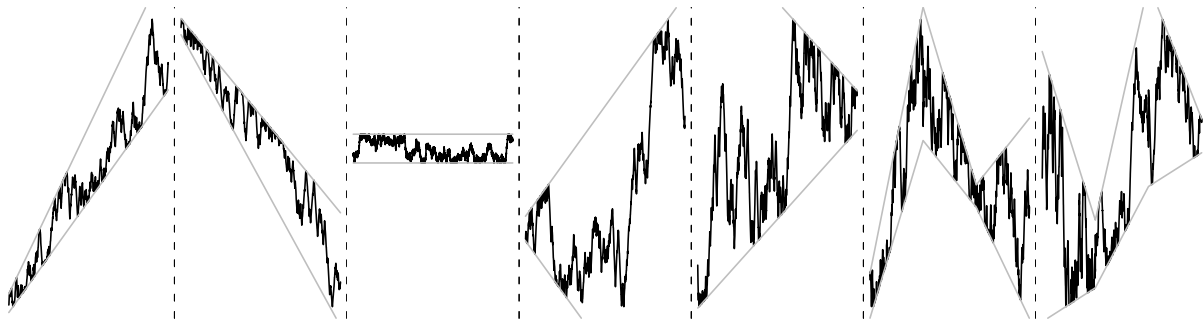


Figure 2: The seven artificial times series problems.

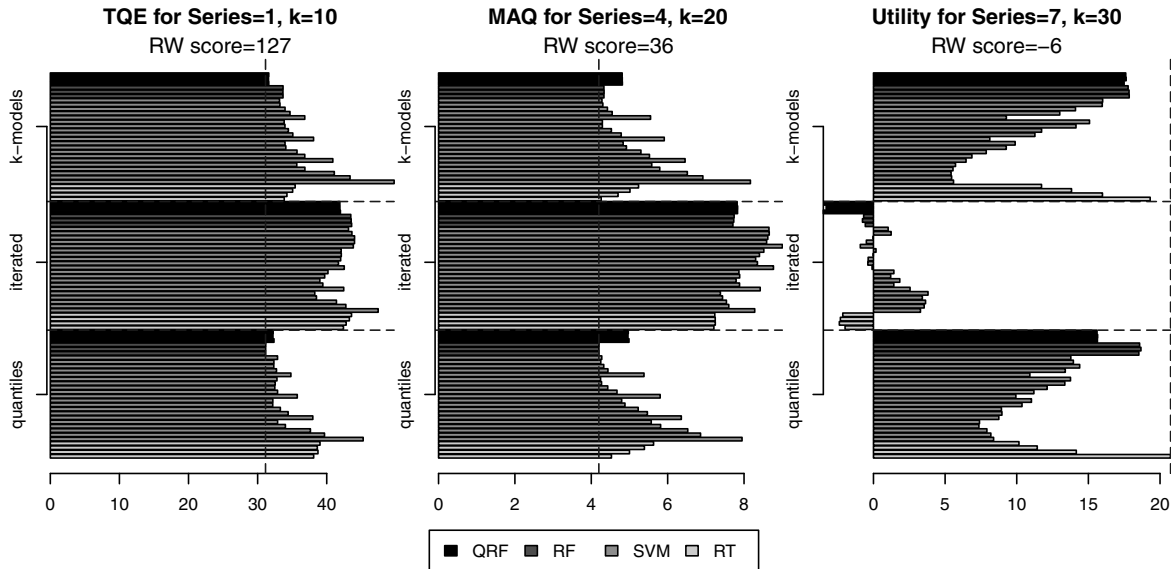


Figure 3: The results on three different experimental setups for the artificial time series.

We have observed that the “k-models” and “quantiles” approaches have rather similar results on these problems. Still, when there is some slight difference this is more frequently favorable to our approach. The “iterated” approach on the contrary is most of the times the worst in terms of scores, although all models typically outperform the baseline *RW* model, as expected. This pattern of results is similar over all experimental setups we have tried with these 7 problems. These results show that the prediction accuracy of our approach is highly competitive with the existing alternatives. Moreover, these scores are obtained with a significant advantage in terms of computational efficiency. In effect, while our approach requires two models to be obtained (one for each quantile), the “k-models” approach requires as many models as the length of the interval, i.e.  $k$  models. This is a significant difference as shown in Figure 4. This figure shows the ratio between computation time of “k-models” over our approach. We can observe that, depending on the size of the interval, the “k-models” approach can take from 5 to 30 times more time to be obtained. On dynamic environments where new data is constantly being collected, eventually requiring new models to be obtained, this margin can be rather significant. We do not show the results for the “iterated” approach as they are essentially similar to our proposal.

Summarizing, the experiments on these artificial problems show that our proposal is able to achieve a rather competitive prediction accuracy with a significantly smaller computational cost.

### 3.3 Experiments with Real World Data

The first real world application of 2D-intervals we describe is related with water quality control in the distribution network of the metropolitan region of Porto, Portugal (serving roughly 1 million people). The company (AdDP) that manages this network has legal limits that must not be reached for several water quality parameters that are monitored. With the goal of avoiding the danger of crossing these limits (and paying the respective high fines) the company internally establishes tighter limits that if broken generate an alarm that leads to several control actions over the network. These actions have associated costs and thus having these internal limits too tight will lead to a high operational cost of the network. However, having wider internal limits, possibly too near the legal ones, will increase the risk of when the alarms are fired being too late for the control actions to have any effect to avoid breaking the legal limits. This means that establishing the intervals of acceptable (normal) values of a large set of water quality parameters is a problem

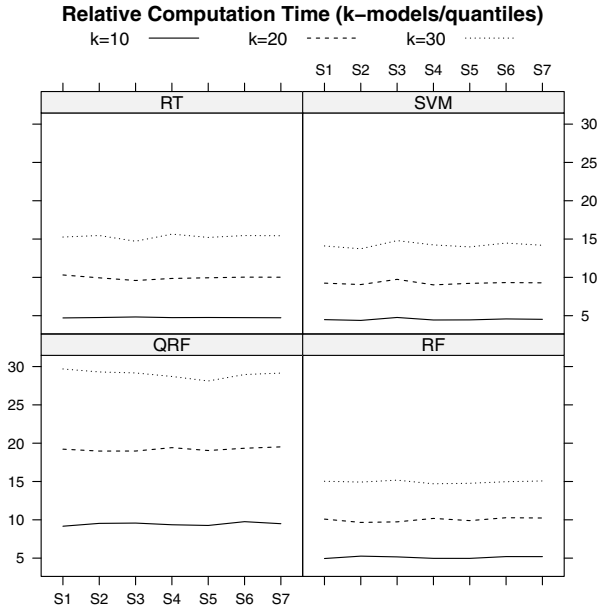


Figure 4: The relative computation times of “k-models” vs “quantiles” on the 7 artificial time series.

with high socio-economical impact for the company and the region. The company is aware of the fact that these intervals that establish a range of normality for each parameter change along the year as the rivers from where the water is collected are very dynamic and change a lot during the seasons of the year. This means that this notion of normal parameter values is dynamic, though the legal limits are fixed.

The AdDP company has provided us daily data concerning a large set of water quality parameters along several years (2000 till 2008). In this paper we focus on the task of trying to forecast a 2D-interval of normal values for a small set of parameters (*pH*, *iron*, *turbidity* and *aluminium*) using this data set. This is a task similar to the ones described in Section 3.3. The interval of “normal” values can again be approximated by the expected 1st and 3rd quartiles on a future time window, i.e. a 2D-interval. We have used exactly the same prediction tasks and predictors as in the artificial problems (c.f. Equation 4). However, following the company requirements, we have only applied our method for a 2D-interval of 30 days (i.e.  $k = 30$ ).

The results we report are again estimates of the 3 evaluation statistics using 10 repetitions of a Monte Carlo simulation. We have used the previous 365 values of each of the four time series and tested the models along a 90 days window, again using a sliding window approach.

Figure 5 shows the results for 3 different setups using the same schema as before. On the left graph we have the *TQE* scores for the *Iron* time series. The central graph illustrates the results in terms of *MAQ* for the *pH* parameter, while the right-most graph shows the *Utility* results for *Turbidity*. The overall pattern is similar to the results on the artificial problems. The “iterated” approach is clearly the worst method, while our approach and “k-models” get similar scores. Still, compared to the results on the artificial data, we observe a more marked advantage of our proposal.

The exception is the *QRF* model variants where the “k-models” approach achieves better results whilst not the best overall scores.

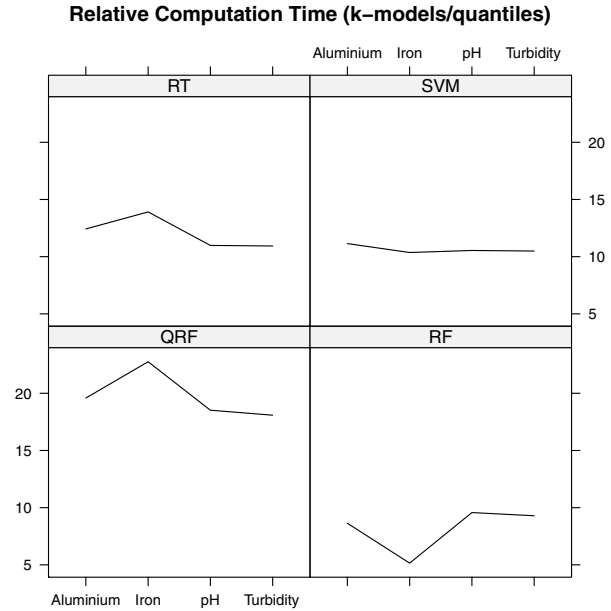


Figure 6: The relative computation times of “k-models” vs “quantiles” on the water quality control problem with  $k = 30$ .

In terms of computation times the results are shown in Figure 6. Again we observe a significant overhead of the “k-models” approach when compared to our proposal.

The second real world problem concerns again a water distribution network, this time in the south of Spain. The problem here is related to production planning in order to face the varying demand in terms of water consumption. We have hourly data concerning the water consumption in a residential area of the water distribution network from Jan, 2005 till Apr, 2005. Our goal is to forecast a 2D-interval for the next 12 and 24 hours (values of  $k$ ). The distribution of the water demand has very marked seasonal properties not only along the different periods of the day, but also across similar weekdays. Because of this we use a slightly different approach in terms of predictors with the goal of providing the models with information on this weekly seasonal effects. For  $k = 12$  we have used the last 6 values of the demand (last 6 hours), the same information as on previous problems regards the quartiles, mean and standard deviation on the past  $k$  values, but also the quartiles that we want to forecast measured in the previous week (i.e. on the same weekday/hour) to provide information on this observed weekly seasonality. In the case of  $k = 24$  we have increased the number of past values of the series from 6 to 12, while the other predictors stayed the same.

Regards the experiments we have used again 10 random repetitions of a Monte Carlo simulation this time with a training window of 1344 values (8 weeks) and a test window of 336 values (2 weeks), for which predictions were obtained using a sliding window approach. The size of the 2D-intervals ( $k$ ) was set to 12 and 24 hours.

The results of 3 different setups of these experiments are

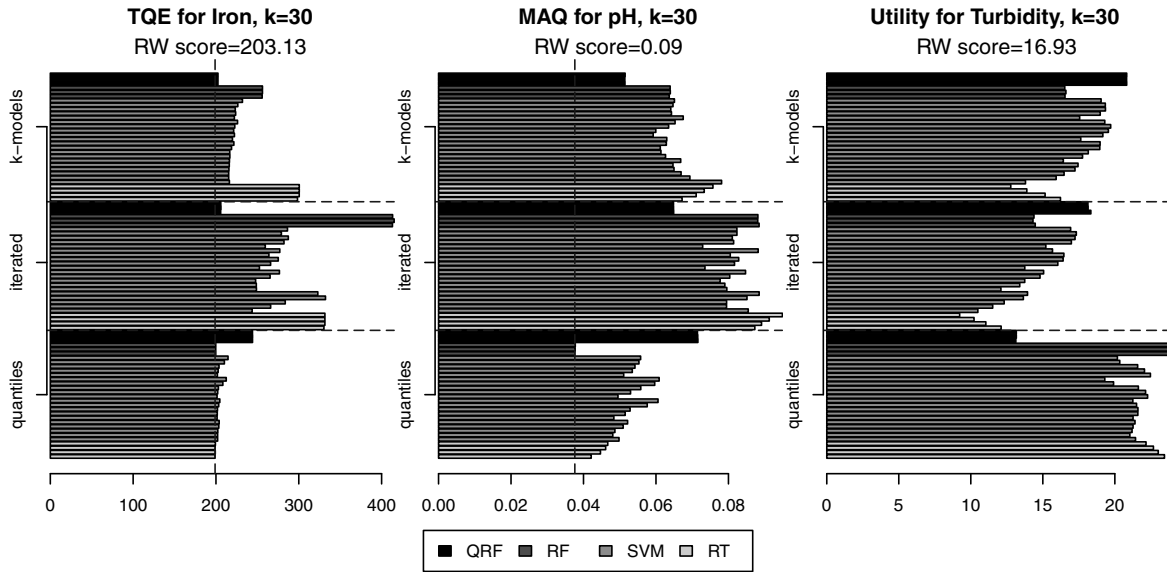


Figure 5: The results on the water quality control problem.

show in Figure 7. The pattern of results is similar to the one observed in the water quality problems. Both “k-models” and our approach have similar results with a slight advantage of our method, while the “iterated” approach clearly lags behind. Once again we have observed that the *QRF* models achieve better results with the “k-models” approach. We should note that the *Utility* score is particularly relevant for this type of applications where serious mistakes may have significant financial costs.

In terms of computation times the scores are shown in Figure 8, and reveal the same type of advantage of our proposal over the “k-models” alternative.

#### 4. CONCLUSIONS

This paper has described 2D-interval predictions, a class of time series tasks with a high relevance for several application domains. To the best of our knowledge there is no established methodology to handle these problems across the several disciplines that address time-dependent data. The main goals of this paper were: (i) to raise the awareness of the data mining community to these relevant problems, and (ii) to propose a new methodology to address these tasks that can be used with any time series modeling technique. The proposal consists in directly predicting the distribution statistics for the target time interval, which allows for instance to obtain a range of plausible values for this period.

We have described our proposal and have carried out an extensive set of experiments designed with the goal of checking the validity of the proposal when compared to existing alternatives. This comparison was carried out from two perspectives: (i) the perspective of prediction accuracy of the 2D-intervals, and (ii) the perspective of the computational cost of the alternatives. This latter issue may be particularly relevant in dynamic contexts where new data arrives at a high pace. The question of the accuracy of the predictions was also addressed from different perspectives trying to capture characteristics that are important to this type of applications (e.g. the costs and benefits of the predictions).

The results of our experiments with several artificial time series and also two real world problems provide clear evidence on the validity of our proposal. It achieves a prediction accuracy that it is highly competitive with the best alternatives in the different experimental setups that were considered, but with a significantly lower computational cost. This makes the proposal particularly adequate for high-frequency time series where 2D-interval predictions may be of use.

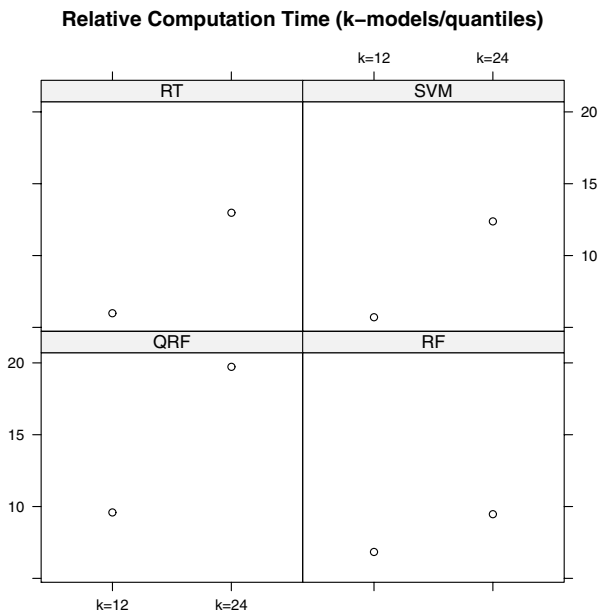


Figure 8: The relative computation times of “k-models” vs “quantiles” on the water consumption problem.

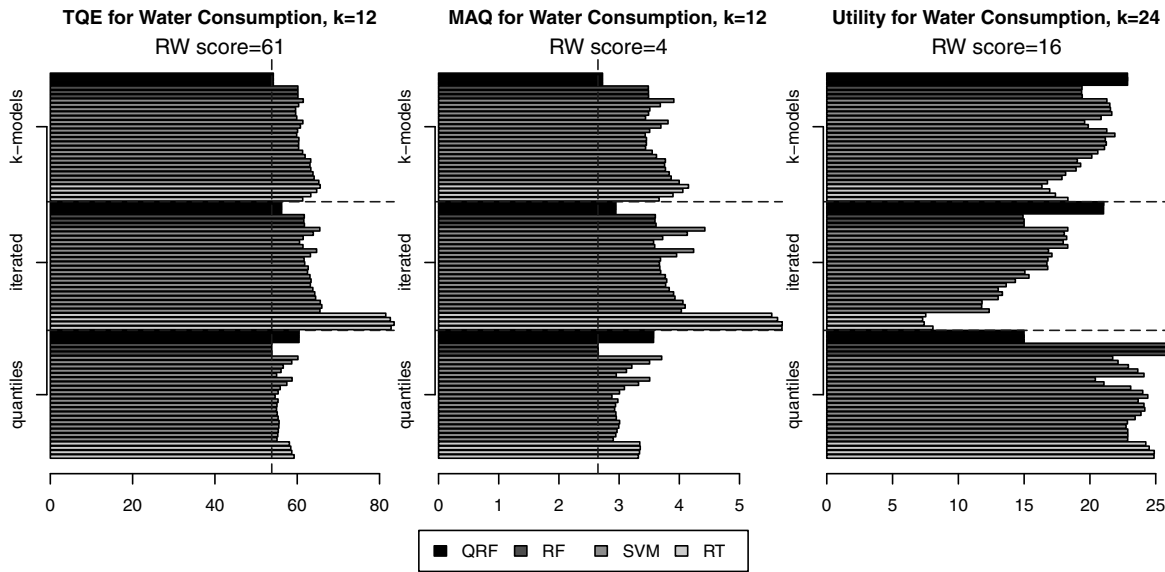


Figure 7: The results on the water consumption problem.

In terms of future work we plan to study our proposal from a more theoretical perspective with the goal of understanding why it achieves these competitive results. Namely, we will try to check the hypothesis that this performance may be caused by the smaller difficulty of the task we address. Predicting these distribution statistics directly, instead of calculating them from the time series predictions, may be easier because distribution statistics (particularly the quantiles we are using) are more stable and not so exposed to the effects of spurious variations on the original time series values.

## 5. REFERENCES

- [1] L. Breiman. *Classification and regression trees*. Chapman & Hall/CRC, 1984.
- [2] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
- [3] J.B. Bremnes. Probabilistic wind power forecasts using local quantile regression. *Wind Energy*, 7(1):47–54, 2004.
- [4] C. Chatfield. Prediction intervals. *Journal of Business and Economic Statistics*, 11:121–135, 1993.
- [5] C. Chatfield. *Time-Series Forecasting*. Chapman & Hall/CRC, 2001.
- [6] C. Chatfield. *The Analysis of Time Series, an introduction*. Chapman & Hall/CRC, 2004.
- [7] N. Cristianini and J. Shawe-Taylor. *An introduction to support Vector Machines: and other kernel-based learning methods*. Cambridge Univ Pr, 2000.
- [8] E. Dimitriadou, K. Hornik, F. Leisch, D. Meyer, and A. Weingessel. *e1071: Misc Functions of the Department of Statistics (e1071), TU Wien*, 2009. R package version 1.5-22.
- [9] C. Elkan. The foundations of cost-sensitive learning. In *International Joint Conference on Artificial Intelligence*, pages 973–978, 2001.
- [10] R. Koenker. *Quantile Regression*. Econometric Society Monograph Series. Cambridge University Press, 2005.
- [11] A. Liaw and M. Wiener. Classification and regression by randomforest. *R News*, 2(3):18–22, 2002.
- [12] N. Meinshausen. Quantile regression forests. *J. Mach. Learn. Res.*, 7:983–999, 2006.
- [13] N. Meinshausen. *quantregForest: Quantile Regression Forests*, 2007. R package version 0.2-2.
- [14] C. Perlich, S. Rosset, R.D. Lawrence, and B. Zadrozny. High-quantile modeling for customer wallet estimation and other applications. *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '07*, pages 977–985, 2007.
- [15] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, 2010.
- [16] A. S. Tay and K. F. Wallis. Density forecasting: a survey. *Journal of Forecasting*, 19(4):235–254, 2000.
- [17] J.W. Taylor. Density forecasting for the efficient balancing of the generation and consumption of electricity. *International Journal of Forecasting*, 22:707–724, 2006.
- [18] J.W. Taylor, P.E. McSharry, and R. Buizza. Wind power density forecasting using ensemble predictions and time series models. *IEEE Transactions on Energy Conversion*, 24:775–782, 2009.
- [19] T. M. Therneau and B. Atkinson. R port by B. Ripley. *rpart: Recursive Partitioning*, 2009. R package version 3.1-44.
- [20] L. Torgo. *Data Mining with R, learning with case studies*. CRC Press, 2010.
- [21] W.B. Wu, Z. Xu, and Y. Wang. Robust prediction of network traffic using quantile regression models. In *2006 IEEE International Conference on Information Reuse and Integration*, pages 220–225, 2006.