EP2420 Project 2 - Forecasting Service Metrics

Rolf Stadler Forough Shahab

November 21, 2020

Project Objective

Forecasting is the process of making predictions of the future based on past and present data. An example is the prediction of a metric of interest at some specified future time. In this project, we forecast service-level metrics of a Video-on-Demand (VoD) service and of a Key-Value (KV) store based on traces collected from a KTH testbed. We take two approaches to forecasting: First, we formulate forecasting as a regression problem and use linear regression and neural network regression to solve it. Second, we use time series analysis to obtain a solution.

Project tasks

You will receive a data set with traces collected from a KTH testbed. The traces originate from running a service (either VOD or KV) on the testbed infrastructure while monitoring device and service metrics. The testbed is described in [1].

For all tasks in this project, apply one of the methods described in Project 1, Task 1 to pre-process a trace and remove possible outliers.

Apply tree-based feature selection to reduce the number of features in the trace to 16. Use 10 trees as hyperparameter.

In the project report, explain the concepts of Long Short-Term Memory (LSTM) and Autoregressive Integrated Moving Average (ARIMA).

Task I - Using linear regression for forecasting

1. For general regression problems, we do not assume the index of a sample (i.e., an observation or an extracted metric) to be ordered. In this project however, we take advantage of the fact that the sample index of a trace $\{(x^{(t)}, y^{(t)})\}$ is a timestamp, which defines a strict order on the samples. This means that all samples form a sequence.

2. For general regression problems, the objective is to find a model (or function) $M(\theta) : x \to \hat{y}$ with parameter θ so that $\hat{y}$ closely approximates y for $x \in \mathbb{R}^n$, $y \in \mathbb{R}$. This is achieved by minimizing a loss function $\mathcal{L}(\{M(\theta; x^{(t)}), y^{(t)}\})$ with respect to θ, given the samples $(x^{(t)}, y^{(t)})$ $t = 1, ..., m$ that are drawn from a hidden distribution $p(x, y)$.

3. For the purpose of forecasting, we formulate the regression problem as follows. For given values of $l, n \in 0, 1, 2, ...$ the objective is to find a model (or function) $M(\theta, h, l) : [x_{-l}, ..., x_0] \to [\hat{y_0}, ..., \hat{y_h}]$ with parameter θ so that $\hat{y} = [\hat{y_0}, ..., \hat{y_h}]$ closely approximates $y = [y_0, ..., y_h]$ for $x \in \mathbb{R}^{n(l+1)}$, $y \in \mathbb{R}^{h+1}$. This is achieved by minimizing a loss function $\mathcal{L}(\{M(\theta, h, l); x^{(t)}, y^{(t)}\})$ with respect to θ, given the samples $(x^{(t)}, y^{(t)})$ $t = 1, ..., m$ that are drawn from a hidden distribution $p(x, y)$. h is called the time horizon, l is called lag.

4. Data preparation: Use one of the methods described in Project 1, Task 1 to pre-process the trace. Remove possible outliers. Reduce the dimensionality of the feature space to $k = 16$ using tree-based feature selection. Then, split the processed trace into training and test samples $(x^{(t)}, y^{(t)})$ by assigning the samples with $t < T$ to the training set and $t \geq T$ to the test set. T is chosen so that the training set contains 70% of the samples.

5. Create a new training set and a new test set with samples of structure $([x^{(t-l)}, ..., x^{(t)}], [y^{(t)}, ..., y^{(t+h)}])$.

6. For the KV service, h is incremented and l is decremented by 1 sec intervals between samples. For the VOD service, h is incremented and l is decremented by 300 sec between samples.

7. Using linear regression, train models for $l = 0, .., 10$ in the feature space and $h = 0, .., 10$ in the target space. The model with $l = 0$ corresponds to prediction using the current sample. A model with $l > 0$ corresponds to learning on l samples into the past and predicting $0, ..., 10$ steps into the future. Evaluate the models by computing the error (NMAE) on the test set. Display the results in a table with rows representing the time horizon $h = 0, .., 10$ and columns representing the lag $l = 0, .., 10$.

8. Discuss the table with the evaluation results and draw conclusions with respect to accurate forecasting on your trace.

Task II - Using Recurrent Neural Networks (RNNs) for forecasting

1. Recurrent Neural Networks (RNNs) allow for sequence-to-sequence learning. If we define the state of system at time t as $(x^{(t)}, y^{(t)})$ and assume that t defines a strict order on the samples, we can use an RNN to map a sequence of input states $(\{x^{(t_i)}, ..., x^{(t_j)}\})$ to sequence of output states $(\{y^{(t_l)}, ..., y^{(t_k)}\})$. An RNN is an example of a sequence-to-sequence architecture. In this project, we will use a specific version of RNN called Long Short-Term Memory (LSTM), which has proved effective for time-series forecasting [2].

2. The goal of this task is to find a non-linear, LSTM model to forecast the target variable at specific points in time. Similar to part 3 of Task I, we define a model on the sequence of inputs x with lag size l which predicts the sequence of outputs with time horizon h. Similar to part 4, 5 and 6 of Task I we create the training and test sets.

3. Similar to step 7 of Task I, for values of $l = 0, ..., 10$ and $h = 0, ..., 10$ train LSTM models. Evaluate the models by computing the error (NMAE) on the test set. Display the results in a table with rows representing the time horizon $h = 0, ..., 10$ and columns representing the lag $l = 0, ..., 10$.

4. Discuss the table with the evaluation results and draw conclusions with respect to accurate forecasting on your trace. Compare the results with those from Task I.

5. Optional: Increase that range for lag and horizon from 10 to 64 as follows: $l = 0, ..., 64$, $h = 0, ..., 64$. For the KV service, h is incremented and l is decremented by 1 sec intervals between samples. For the VOD service, h is incremented and l is decremented by 60 sec between samples. Display the results in a table with rows representing the time horizon $h = 0, 2, 4, ..., 64$ and columns representing the lag $h = 0, 2, 4, ..., 64$. Comment on the results.

Task III - Time series analysis for forecasting

1. In this task, we apply traditional univariate forecasting methods. This means we only consider the target values $y^{(t)}$ of the trace and do not consider the input values $x^{(t)}$.

2. Perform forecasting using Autoregression (AR). AR models the next step in the sequence as a linear function of the observations at previous time steps. The method is suitable for univariate time series without trend and seasonal components. For the evaluation, set the AR model parameter $p = 1, ..., 10$.

3. Perform forecasting using Moving Average (MA). MA models the next step in the sequence as a linear function of the residual errors from a mean process at previous time steps. Note that MA is different from calculating the moving average of the time series. The method is suitable for univariate time series without trend and seasonal components. Set the model parameter $q = 1, ..., 10$.

4. Perform forecasting using Autoregressive Integrated Moving Average (ARIMA). ARIMA models the next step in the sequence as a linear function of the differenced observations and residual errors at previous time steps. It combines both AR and MA including a differencing pre-processing step of the sequence to make the sequence stationary, called integration (I). ARIMA is suitable for univariate time series with trend and without seasonal components. Set model parameters (p,d,q) as follows: $d = 1$, $p = q = 1, ..., 10$.

5. In all above cases, for the KV service, h is incremented and p, q are decremented by 1 sec intervals between samples. For the VOD service, h is incremented and p, q are decremented by 300 sec between samples.

6. Evaluate the three methods AR, MA, and ARIMA in the same way as the forecasting method in Task 1. Analyze the results and draw conclusions with respect to suitability of the methods, parameters and accuracy of forecasting on your trace.

   A detailed introduction to time-series analysis and forecasting can be found in [3].

References

[1] F. S. Samani, H. Zhang, and R. Stadler, "Efficient learning on high-dimensional operational data," in *2019 15th International Conference on Network and Service Management (CNSM)*, IEEE, 2019.

[2] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016.

[3] C. Chatfield, *The analysis of time series: an introduction*. Chapman and Hall/CRC, 2003.