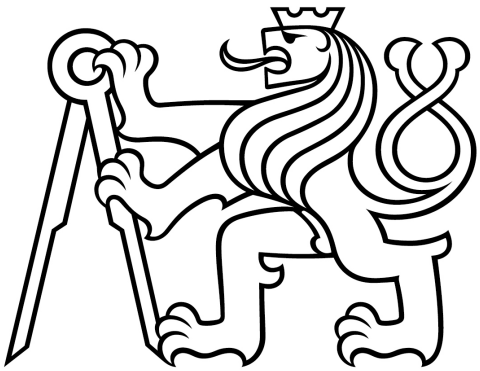
Czech Technical University
Faculty of Electrical Engineering
Department of Cybernetics



Bachelor's thesis

Active learning for prediction of continuous variables

Matěj Niederle

Supervisor: Ing. Macaš Martin, Ph.D

Study Programme: Open Informatics
Field of Study: Computer and Informatic Science
January 2020

BACHELOR'S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Niederle Matěj** Personal ID number: **456888**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Cybernetics**

Study program: **Open Informatics**

Branch of study: **Computer and Information Science**

II. Bachelor's thesis details

Bachelor's thesis title in English:

Active Learning for Prediction of Continuous Variables

Bachelor's thesis title in Czech:

Aktivní učení pro predikci spojitých proměnných

Guidelines:

1. Propose an active learning strategy for de-novo synthesis of training set for prediction of continuous variables.
2. Implement, test and analyze the proposed strategy on chosen synthetic regression tasks.
3. Implement, test and analyze the proposed strategy on a chosen time-series forecasting task.
4. Analyze if such a method can bring some benefits in comparison to random sampling strategy.

Bibliography / sources:

[1] BURBIDGE, Robert; ROWLAND, Jem J.; KING, Ross D. Active learning for regression based on query by committee. In: International Conference on Intelligent Data Engineering and Automated Learning., 2007. p. 209-218..
[2] WILLETT, Rebecca; NOWAK, Robert; CASTRO, Rui M. Faster rates in regression via active learning. In: Advances in Neural Information Processing Systems. 2006. p. 179-186.
[3] MACAS, Martin, et al. The role of data sample size and dimensionality in neural network based forecasting of building heating related variables. Energy and Buildings, 2016, 111: 299-310.

Name and workplace of bachelor's thesis supervisor:

Ing. Martin Macaš, Ph.D., Cognitive Neurosciences, CIIRC

Name and workplace of second bachelor's thesis supervisor or consultant:

Date of bachelor's thesis assignment: **11.01.2019** Deadline for bachelor thesis submission: **07.01.2020**

Assignment valid until: **30.09.2020**

Ing. Martin Macaš, Ph.D.	doc. Ing. Tomáš Svoboda, Ph.D.	prof. Ing. Pavel Ripka, CSc.
Supervisor's signature	Head of department's signature	Dean's signature

III. Assignment receipt

The student acknowledges that the bachelor's thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the bachelor's thesis, the author must state the names of consultants and include a list of references.

.	
Date of assignment receipt	Student's signature

Acknowledgement

I would like to thank Ing. Macaš Martin Ph.D for his guidance with this thesis. Great thanks also has to go to my family who believed in me and friend Han for giving me huge support.

Author's statement

I declare that the presented work was developed independently and that I have listed all sources of information used within it in accordance with the methodical instructions for observing the ethical principles in the preparation of university theses.

Prague, date ....

Abstrakt

Při značném kvantu dat ve světě je potřeba obracet se na metody, které by se zaměřovaly na jejich kvalitu. Tato bakalářská práce se věnuje metodě query by committee, která dokáže zvážit a vybrat data která nejvíce zvýší efektivitu. Tato práce je založená na reálném projektu, který se zaměruje na prediktivní model pro prediktivní kontrolu vytápění v kancelářské budově. Bakalářská práce zkoumá, zda generování optimálních setpointů teploty pro regresní prediktivní model zlepšuje efektivitu předpovědi a labelování. Po zhotovení experimentů se ukázalo, že tato metoda nepředčila originální strategii použitou v původním projektu. Možné příčiny takového výsledku jsou později diskutovány.

Klíčová slova: aktivní učení, query by comitee, predikce

Abstract

The size of data in today's modern world has urged people to resort to strategies that focus on the quality of data. This thesis revolves around a method called query by committee that is able to consider and choose what data it needs to be the most effective. This thesis is based on a real world problem that is related to the predictive model for predictive control of heating in an office building. Here, the focus is to examine whether generating an optimal temperature setpoints for the regression based predictive model for the control of a heating plant improves the forecasting efficiency and reduces the labeling process. The conducted experiments demonstrate that this method does not manage to outperform the original strategy used in the original problem and a discussion is held on possible reasons why.

Keywords: active learning, query by committee, prediction

Contents

List of Figures

List of Tables

Acronyms

AUTC Area under the training curve. 10, 11, 14, 17–20

EM Expectation-Maximization. 3

HAMBASE "Heat, Air and Moisture model for Building and System Evaluation". 15–17

KQBC Kernel Query by Committee. 3

MAPE Mean Absolute Percentage Error. 17, 18, 20

MSE Mean Squared Error. 10–14

NN Neural Network. 2, 20, 21, 23

QBC Query by Committee. 1–7, 10, 11, 13, 14, 16–19, 21, 23

Chapter 1

Introduction

1.1 Motivation

Technological progress of today's world has allowed people to collect huge amount of data. Such advance has created an environment suitable for the use of many machine learning algorithms since the limited power is no longer an unavoidable obstacle. However, it is costly to process such amounts of data, which restricts the use of algorithms once again. The need to work with such a huge amount of data pushed us to find ways how to reduce the data size and focus on quality of data, not their size. One approach is to go through the set and choose the optimal data during learning, which is called active learning.

Active learning has many strategies to conclude which data points should be labelled (to determine a value of an instance of data), which can be categorized into certain methods. The particular focus of this thesis is a method called QBC. QBC is a method proposed by Seung, Opper and Sompolinksy in [1] that creates a committee of learners which are taught on collected data. The selection of the next instance to be labeled (technically called query) is based on where committee members' disagreement is the largest, an approach called the principle of maximal disagreement.

It should be noted that part of this work is related to the predictive model for predictive control of heating in an office building. This real world problem is described in [2]. During the process of identification data acquisition, the input variables (temperature setpoints) are preset randomly, which leads to diverse queries but not optimal ones in terms of a proper excitation, resulting in a need of bigger training data, longer training times, lower precision.

Although the active learning can reduce the amount of data needed for learning, we can also use it to generate optimal data. This approach is called query synthesis de novo [3] and it is an approach used in this thesis. It was also used for a regression learning task, where the absolute coordinates of a robot hand was predicted based on the joint

angles of its mechanical arm [4]. We use it to generate optimal temperature setpoints for the regression based predictive model for the control of a heating plant. Our objective is to examine whether this method is able to enhance the forecasting efficiency, reduce the labeled data set and shorten labeling process overall.

1.2 Objectives

This thesis consists of four main objectives. First, we propose an active learning strategy that can be used for construction of a training set for prediction of continuous variables. The proposed strategy will be used later in all experiments.

The second objective is to implement, test and analyze the proposed strategy on a simple, yet informative synthetic regression task. This is done to examine if our proposed strategy can be used later on – if the proposed strategy works and has any chance to enhance the forecasting task.

After the synthetic regression task comes the primary task of this thesis – time-series forecasting task. In this objective, we use the proposed strategy and compare it to the strategy used in the original work [2].

Finally, we will analyze if there are any benefits in using the proposed strategy over the original one.

1.3 State of the art

Some fields such as astronomy have labels that are very costly to compute as was mentioned in [5] while presenting the use of active learning to lessen the negative effects of constraining parameters of the physical model. Both QBC and Query by Dropout Committee are used, showing that both permit the opportunity improve efficiency of the parameter constrain and so it offers better results than common sampling algorithms that are currently used.

Active learning and QBC have been utilized for classification in [6] to speed up Quantum Few-Body calculations. The calculations face difficulties due to the issue of determining a multi-dimensional function, a known problem within the scientific community. The paper specifically uses Quantum Three-boson problem to illustrate the sped up process, applying different Neural Networks (NNs) as a committee.

Authors of [7] have applied QBC for regression in the development of surrogates as physics-based earthquake ground-motion simulators. Again, NNs have been used as an example of surrogates due to their competency in challenging model estimations. The results of the generalization error showed that the active learning approach was better than passive

learning, with the same amount of training data. It is important to note that although this study is limited to one earthquake and one metric, it brought an interesting insight to surrogates as physics-based earthquake ground-motion simulators.

Paper [8] introduced an improvement of a sampling strategy for QBC based on inconsistency ranking for gas sensor array signal processing. This approach rates the query data corresponding to the discrepancy in the committee vote results and selects a particular number of samples at the top at once. The experiments demonstrated that this method needed a small number of initial training samples while the accuracy dramatically improved after adding only few actively selected samples.

An issue with long periods of training data collection for the user before operating the system was mentioned in a brain-computer interface related paper [9]. To reduce the amount of training data while maintaining the performance, QBC method is utilized, forming the committees in heterogeneous and homogeneous feature spaces. Especially, the QBC with heterogeneous feature space has decreased the cost of labelling notably well.

Since QBC is simple and effective algorithm, it influenced a creation of other algorithms. One example of that is an algorithm named Kernel Query by Committee (KQBC), introduced in [10]. Although QBC does indeed lower the cost of training learning algorithms, its sampling step from high dimensional version space is well-known to be demanding. KQBC samples from low dimension spaces, enabling an option to manage large scale problems. Due to that, KQBC also allows the utilization of kernels for non-linear situations hence its name.

An alternation of QBC has been introduced in [11] for text classification, using Expectation-Maximization (EM). The modification lowered the amount of needed labelled training data by utilizing the unlabeled pool to estimate density when picking examples for labelling. The method then applies EM algorithm for the rest of the class labels that stayed unlabelled. The combination of EM and active learning has positively affected the amount of needed labelled training data and provided satisfactory results.

All in all, QBC is still used nowadays with more and more applications for it. While classification problems solved by QBC are in majority, regression based tasks are not rarity at all. Be it robot arm position prediction, earthquake simulators or this thesis, heating plant predictive control, regression based QBC is still well alive today.

1.4 Structure

The structure of this thesis closely follows the already described guidelines of the project. For the theoretical part, chapter 2 covers basic theory that is needed for further understanding of the thesis: the difference between classification and prediction, active learning and the main focus – QBC. The last section 2.3 is dedicated to the description of the proposed strategy based on QBC.

Before proceeding to the main task, the complex forecasting task, a simple curve-fitting problem is presented in the following chapter 3. It is done so with the intention of verification of the proposed strategy. The purpose of this is to find whether the proposed strategy can even be used and perhaps, the effectiveness of such strategy. A few conducted experiments are then described and results presented. Ultimately, a discussion is held to point out the possible imperfections of the proposed strategy.

Chapter 4 revolves around the main objective of the thesis. After introducing the original prediction task that we are attempting to improve, the necessary information about the considered building, data acquirement through simulation and predictors are described in this respective order. The crucial part of this chapter is section 3.2 that talks about the conducted experiments and their results. Again, a discussion is held to further contemplate about it.

Lastly, chapter 5 drew a conclusion about thesis.

Chapter 2

QBC Active Learning

This chapter is essential to the reader's basic understanding of the following chapters of the thesis. It describes prediction, introduces active learning and QBC as its particular instance. Since active learning is commonly used for classification, it is also important to clearly distinguish the difference between classification and prediction.

It is important to note, that this work's main theme is prediction of continuous variables and not classification. These two approaches share many similarities, where prediction might be more of a common term and deciding an output is how it is differentiated. When output is an established discrete class, it is called a classification. When the output is a continuous variable, it is regression and lastly, when we actually try to foresee/predict a variable in a future, it is forecasting. This thesis uses the term prediction with meaning of the regression.

In other words, while the target output of the model in classification task is a discrete variable (class label), the prediction model outputs a continuous value [12]. A real world example of this is the following – classification is used to determine if an item on an apple tree is a leaf or an apple, while prediction would be how many apples is on the tree.

2.1 Active learning for prediction

Machines learn in similar way any living organism does. It needs to see a lot of objects, be told what they are and remember it. This thesis examines the first two steps, where we need to gather sufficient amount of objects and name them. These objects and their names form a data, that are a substantial part of the learning process. Data represents inputs and outputs of models and are used for a construction of machine learning prediction or classification model. Depending on the principle of gathering these object (data), we discern two approaches, passive and active learning.

Passive learning represents gathering a large amount of data and using them to train a model for the needed machine learning task. Gathering and labeling a large amount of data is usually time-consuming, it takes most time of the process just to arrange them, but the learning on them itself is effective.

However, there are situations when collecting a large data set is not suitable due to various reasons, such as high cost of labeling, little amount of sources etc. In such cases, the need to be able to filter, predict or sometimes even generate data in such way, that learning algorithm needs as few data points as possible.

Process where we estimate next data point to label based on data collected so far is called active learning. It is important to note, that although it is typically used in the pattern classification domain, our goal is to apply its principles to the regression domain.

Active learning is a machine learning method where the algorithm itself chooses the data it deems necessary in order to accomplish its task. Since it has the ability to make informed decisions regarding selecting new instances, active learning algorithms tend to need substantially less training data than the traditional methods [13].

Let us make an example where active learning is relevant. Suppose there is a hospital, where a new experimental treatment is being performed and we want to predict the ratio of success. We can post a recruitment and accept first 100 people who respond, but random element will be in place and we might end up with uneven testing sample – such as most participants will be students under 25 years old. In worst cases, doctors determine treatment effectively, but later, it will show no effects for elderly. Instead of that, we can look for people one by one, based on who we already have. If first 5 people are students under 25 years old, no more student will be accepted and another age group is looked for. This way, more balanced testing will be performed and even number of people invited might be less.

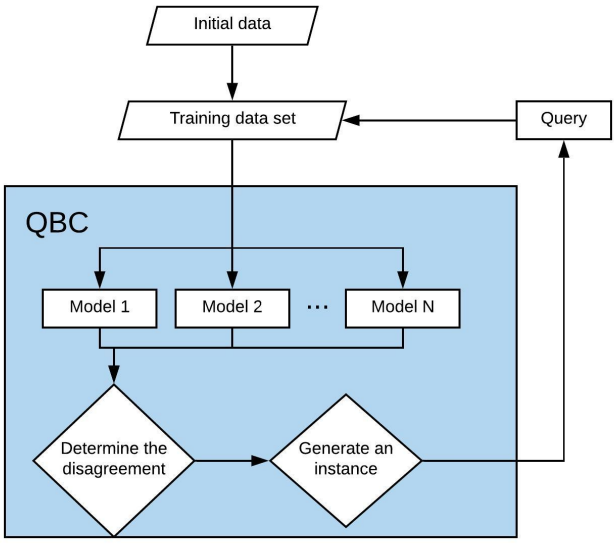


Figure 2.a: Active learning diagram with QBC

The process where we try to actively look for the next person (or query for an instance) is what is called active learning. Let us look at a simple diagram representing active learning

in Fig. 2.a. We start with some initial data that is transformed into the training data set. This training data set is then given to a query selection strategy (query selection is explained in the chapter 2.2) which gives us an instance to query. In the end, the labeled instance is added into the training data set.

2.2 Query by committee

Most commonly used approach for data gathering in active learning is called pool-based sampling, where machine learning algorithm has a large pool of unlabeled data where it can choose which it needs. In our case, such a large pool is not available and so we found another approach, called membership query synthesis [3].

In membership query synthesis, the learner is given an input space, from where he can request any unlabeled instance to be queried, which are generally queries that the learner generates de novo. Such queries usually carry the most informative value.

In this thesis, it is presumed that the cost of gaining label y for query x is costly, therefore our objective is to limit querying of the labels as much as possible.

Generally, query by committee is a query selection strategy, but we attempt to use its principles for query synthesis using membership query synthesis strategy. Let there be a labeled data set L, that serves as the initial training data set. QBC maintains a so called membership committee, where the committee votes for an instance to be generated. Each member of the committee should give varying votes, as optimal instance is selected as the one with the highest disagreement among committee members. This principle is illustrated in a Fig. 2.a, highlighted in the blue box.

2.3 Proposed strategy

This thesis focuses on the proposed strategy that is built on QBC. We are given a space from which instances can be queried by an interval for a given variable. The first few instances are chosen randomly from a given space to create an initial data set that is later used for all the experiments.

In our case, we represent the commmittee as a set of diverse regression models of the same concept. To keep the diversity of the models, random subsets of the labeled data are used for the training of each model. Number of committee members, size of initial set of labeled data and variable constraints are modifiable parameters that we will examine in this thesis.

We propose to evaluate the disagreement among committee members using standard deviation of their responses. If all the models provide the same prediction, there is no

disagreement and the standard deviation is zero. If the model responses are different, the standard deviation is non-zero and estimates the level of disagreement.

In a pool based sampling, a training instance with the highest committee disagreement would be selected for labeling among a set of unlabeled instances. In our scenario, such data instance is generated de novo, which maximizes the disagreement, which corresponds to an optimization task, where fitness function is the committee disagreement and its arguments are the values for predictor's input variables.

Chapter 3

QBC for curve fitting

3.1 Curve fitting

Instead of promptly resolving the complex forecasting task, which will be focused later in this thesis, let us verify our proposition on a simple curve-fitting problem to verify whether the proposed strategy can be used and has a chance to be effective. For that reason, we define our testing task as an approximation of a polynomial curve. Whether this method can be an effective solution for this particular problem or not is not our main concern, since the objective of this chapter is solely the comparison against random querying.

Our task in this experiment is to fit a polynomial regression model on the given curve. Curve is represented by a function f:

$$f(x) = 1 - 5sin(x) + 0.0001x^2 - x$$

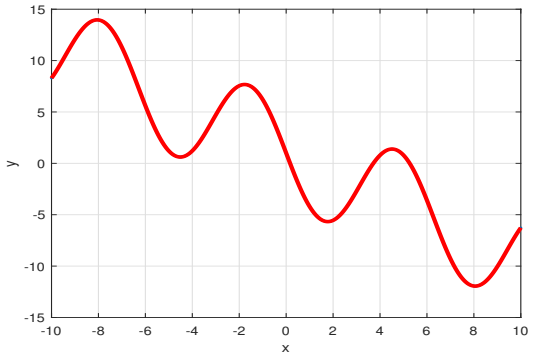


Figure 3.a: Tested function

This curve is plotted in Fig. 3.a. Curve is complex enough for regression polynomial model to have some shortcomings, but it should still be able to give us satisfying results. Examined range of the curve is from -10 to 10.

Committee consists of several polynomials fitted on a random subsets of labeled data. Main variables that we need to control are size of a committee, maximum degree of fitted polynomials and size of initial data (queried randomly).

Query generation is handled by MATLAB[14] function fminbnd, based on golden section search and parabolic interpolation. fminbnd find an extreme of function, in our case

maximum disagreement of the committee (maximum standard deviation). Discovered instance is then processed by function f and added into the training data set.

3.2 Experiments

First issue to solve is the size of the initial set and the number of committee members. Guyon, I. et al. [15] mentions the importance of size of the initial data set as a part of their work. Sometimes having a large initial data set is a viable strategy, however, we need to be able to obtain it. As we obtain initial data set via random querying, obtaining large initial data set would diminish the purpose of this test as we try to compare random querying with QBC.

As for the number of committee members, adjustment to the committee size is done with respect to the size of the initial set, the current size of labeled set each iteration and the length of training. The size of the committee is constant through the whole duration of learning, therefore we need to find a compromise, where large committee will not yield compelling results when we have too few points to learn on, which results in members to not be disagreeing enough. On the other hand, when we have too few members of the committee, in later iterations, the disagreement of members might be biased towards the random subset of training set each member is given.

The last experiment is finally using QBC for querying itself. We experiment with various polynomial degrees of both, various regression models used for the committee and final prediction model.

Expectations are that the higher the complexity is, the more precisely we can model the expected curve, but at the cost of huge difference at the beginning. At the same time, higher polynomial degree can easily end up over-fitting our prediction function which is overall not wanted for the generalization. On the other hand, with polynomial degree being too low, prediction model will not be able to successfully approximate the target curve.

3.3 Results

Quality metric we chose for this experiment is MSE:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

where y_i is a label for instance x_i, $\hat{y}_i$ is an estimated label for instance x_i and n is the total number of instances. Instances were sampled uniformly across the examined interval. We can create our evaluation criteria from MSE – AUTC that represents sum of MSEs over

all iterations (Area under the curve where x axis is number of iterations and y axis is MSE), number of active learning iterations before MSE reaches a certain threshold and from that final savings comparison between random query strategy and QBC

AUTC seems to be an obvious criteria but because MSE of a few starting steps is very high and variable (so called cold-start problem), it mainly points at a general speed of MSE decrease. For that reason, there is also shown AUTC in Tab. 3.1 without first 10 steps of learning algorithm which helps to balance error generated from fitting polynomial with small fitting set.

Degree	Strategy	AUTC	AUTC w/o first 10	Step count	Savings
50	committee	4.34e41± 6.81e42	2.54e22±5.21e19	50.64± 2.2	94%
	random	8.19e90± 1.83e92	3.74e44± 8.38e45	53.7± 6.44	
10	committee	4.72e164± Inf	3.01e68± 6.55e69	23.94± 5.6	53%
	random	2.76e15± 2.84e16	2.89e6± 2.93e7	45.06± 31.23	
5	committee	4.49e50± 1.00e52	2846.2± 6089.2	14.90± 5.45	71%
	random	8.01e6± 4.30e7	9507.5± 51155	20.84± 14.07	

Table 3.1: Results of simulation with 500 repeats and 200 steps.

Values in Tab. 3.1 are all average values from 500 runs with standard deviation shown after ± symbol. Experiment went on with 200 iterations before it stopped, although a terminating metric, such as absolute difference of MSE values from last two iterations reached a needed accuracy, would be implemented for practical use.

The column "Degree" in Tab. 3.1 represents maximum possible degree for models used in committee (exact degree has been chosen randomly every time) and the exact degree used for prediction model. The best results were achieved for the prediction model of degree 10, which is the most similar to our objective function. Savings for polynomials of degree five were not that far behind, but they still fared much better than polynomials of degree 50 where the QBC did not gain much advantage over the random querying.

The first two experiments were essentially similar. Based on one experiment, we tune the other experiment as we look for something efficient, but still quick enough. With most simulations reaching optimums in 20 steps (Tab. 3.1), having an initial data set of size 10 seems to be excessive as almost a half of the final data set is queried randomly. Eventual size of the initial data set has been set to five.

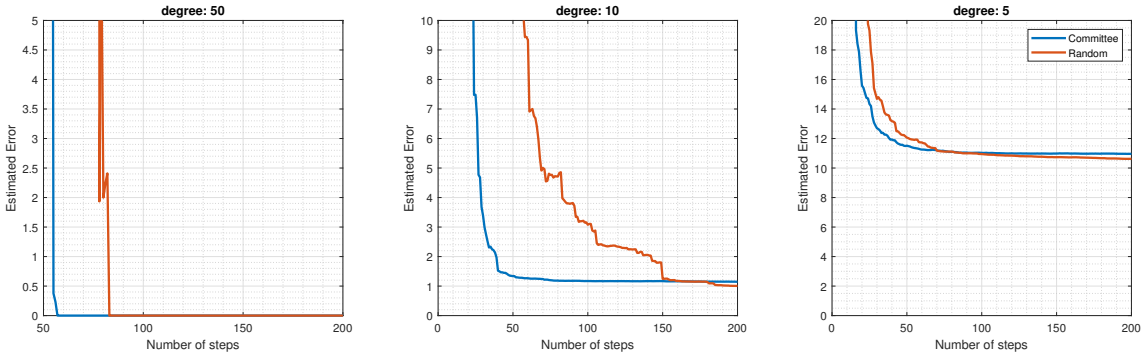


Figure 3.b: Curve fitting experiment MSE results

In Fig. 3.a we see that our function is a combination of a polynomial and sinus functions. For that reason fitting polynomial of degree 5 is practically impossible, as can be seen from the MSE in Fig. 3.b. While the fitting reaches minimal MSE quickly, its final MSE is still high, but that is mostly the case in under-fitting. With higher polynomial degrees, we get lower final MSE, but it takes more iterations, as prediction function is easily over-fitted at the beginning. In the end, It is still better to have some idea about the trend of our forecasting function to better fit our model.

A polynomial regression of one variable might not be very time demanding, but it is still able to give us satisfying results. The few first iterations 3.b did not reach the most satisfying accuracy, but once the training data set has been augmented, the testing error went quickly down. The number of committee members has been settled on four, because experiments with any more members might have ended slightly faster, but the training time overall has increased.

This experiment finished successfully and demonstrated that our proposed method might work. We achieved an increased efficiency of almost 50%, but we had very specific conditions, e.g., objective function was known and we could use a prediction model that closely resembled it or working with only one variable regression.

3.4 Discussion

1. **Optimal threshold of MSE**

The first problem observed is determining the value of a threshold, or how to determine when optimal prediction function is found. One way, the one used in this task is determining threshold after the algorithm is finished. We observe data and simply determine threshold ourselves. This method serves quite well for our purpose since we just want to determine the effectiveness of a method. A practical method to determine threshold might be to watch MSE and stop when absolute difference between new and old MSE is withing given limit.

2. **Starting MSE**

Secondly, it was already mentioned that the error rate is quite high in the few starting steps. The only way to reduce this is to set up a larger initial data set, but as it is done by random sampling, we want initial set as small as possible. Problem with committee is in the size of an initial set and the number of committee members. When committee members outnumber the size of initial data, we get duplicate results as random subsets can (and probably will) be chosen multiple times in one iteration. That eventually affects deviation of committee members, and in the worst case scenario, deviation is constant. When we get constant deviation, then it depends on our optimization function what points get chosen; Even if it is not random choice, it definitely can not be considered a valid point according to definition. This problem does not occur very much in this simple task, but the issue might be very severe with multiple variables.

3. **Variable boundaries**

Another problem lies within setting up bounds. In practice, we have some idea about constraints that our variables should follow (a patient in a hospital has height in range from 1 to 2.5 meters or a turning angle of a joint is from 0 to 90 degrees), but those can still be pretty widespread. Our trouble lies within finding the maximum deviation between committee members. Before the prediction function stabilizes a bit, QBC very often selects border points. No method is effective when it receives identical points all the time – it will ultimately fail to make any progress as subsets selected for the committee contain only these identical points. This problem does not appear with fitting methods that are more complex, used as committee members. However, this issue should always be considered and watched out for.

4. **Model variation viability**

Last problem is not exactly a complication – it is more of an observed occurrence that we did not think of in this easy experiment, but can be demonstrated here for simplicity. Members of committee consist of different regression methods, but not any regression method can be used. Linear regression methods cause query to always select border instance. In the best case scenario, members are all parallel (their disagreement is constant on the entire interval) so query is chosen according to minimizing function. Otherwise, one of the borderline instances is selected.

Chapter 4

QBC for time-series prediction

Primary experiment of this work is to test QBC prediction of continuous variables on time-series forecasting task, previously done in [2]. The strategy in [2] was a random querying strategy for predicting heating-related variables in large office building. The objective is an attempt to increase performance of the prediction by using QBC for querying instances.

Ideal outcome of this experiment lies within predictive control. Predictive control is an optimal-control based method to select control inputs by minimizing an objective function [16]. However, that lies out of the scope of this thesis, therefore outcome of this experiment will be a prediction of the consumption of a heating plant in an office building, on which further control methods could be built.

4.1 Building

Target building that we consider further in this work is modeled after a real office one located at Casaccia Research Centre of Italian National Agency for New Technologies, Energy and Sustainable Economic Development (ENEA). It has a total amount of three above ground floors and there is a thermal plant placed in the basement. There are 41 office rooms with floor areas ranging from 14 up to 36 m^2, two rooms for specialized data processing with $20m^2$ each, four laboratories, one control room and two conference rooms.

Figure 4.a: Front view of ENEA building

Offices are mostly used by two employees. All rooms, laboratories and places are equipped with fan-coils controlled by a proper thermostat that are used as thermal exchangers.

Basement located thermal plant consists of natural gas burner for the winter use and three electronic compressor chillers for the summer use. This experiment is focused on forecasting gas consumption in winter, therefore electronic heaters are later omitted. Data is collected via monitoring system that manages all internal and external environmental sensors and energy consumption. While we are able to obtain data this way, we prefer using simulated data for larger diversity due to the reason that it is highly demanding to collect data with thermal plant set to more extreme temperatures without inconveniencing building residents. [2]

4.2 Simulation

Experimental data for training and testing were obtained with a Matlab Simulink[17] simulator used in the original experiment [2], namely "Heat, Air and Moisture model for Building and System Evaluation" (HAMBASE) [18], [19]. With respect to the sun exposure and thermal changes in each room, building was partitioned into 15 sections for easier computation. Each section connects rooms with similar technical characteristics and thermal conditions. For experiment purposes, we assume only 10 of these 15 sections, where rooms in one sections share one thermostat setting.

While simulation has more outputs, we only work with gas consumption. It is obtained from three aspects, natural gas flow (received from discharge), water temperature in the thermal plant and heating system and last, thermal plant efficiency.

4.3 Predictors

HAMBASE simulator used in original experiment is quite complicated, with 19 inputs. Prediction takes turns in 12-hour intervals, representing main decision making for whole 12-hour period done either in the morning or in the evening. One heating season corresponds to 68 days, therefore we get 134 data instances (Start and end of measuring season is 7AM). List of all variables are shown in Table 4.1.

Number		Description
1	$S_A(t+12)$	Air temperature set point in zones $[^\circ C]$
2	$S_W(t+12)$	Supply water temperature set point $[^\circ C]$
3	$W_1(t)$	Diffuse solar radiation $[Wm^{-2}]$
4	$W_2(t)$	Exterior air temperatures $[^\circ C]$
5	$W_3(t)$	Direct solar radiation $[Wm^{-2}]$
6	$W_4(t)$	Cloud cover (1..8)
7	$W_5(t)$	Relative humidity outside [%]
8	$W_6(t)$	Wind velocity $[ms^{-1}]$
9	$W_7(t)$	Wind direction [degrees]
10	$T_1(t)$	Air temperature in zone 1 $[^\circ C]$
11	$T_2(t)$	Air temperature in zone 2 $[^\circ C]$
12	$T_3(t)$	Air temperature in zone 3 $[^\circ C]$
13	$T_4(t)$	Air temperature in zone 4 $[^\circ C]$
14	$T_5(t)$	Air temperature in zone 5 $[^\circ C]$
15	$T_6(t)$	Air temperature in zone 6 $[^\circ C]$
16	$T_7(t)$	Air temperature in zone 7 $[^\circ C]$
17	$T_8(t)$	Air temperature in zone 8 $[^\circ C]$
18	$T_9(t)$	Air temperature in zone 9 $[^\circ C]$
19	$T_{10}(t)$	Air temperature in zone 10 $[^\circ C]$

Table 4.1: The list of input variables for simulation

Input variables that we primarily focus correspond to the control variables air temperature set point $S_A(t)$ and supply water temperature set point $S_W(t)$. For simplicity, we consider $S_A(t)$ to be held constant during the whole 12-hour interval and changed only before new interval begins.

Other variables required by simulator are as follows:

- Let $t_i(t)$ be the air temperature taken inside a zone i at the end of hour t.
- Let $w_i(t)$ be the various weather measurements taken at the end of hour t. Description of these variables can be found in table 4.1.
- Variable $T_i(t)$ is the average of the 12-hour interval of air temperature in zone i.

$$T_i(t) = \frac{1}{12} \sum_{n=t-11}^{t} t_i(n)$$

- Variable $W_i(t)$ is the average of the 12-hour interval of weather variables $w_i(t)$ described before.

$$W_i(t) = \frac{1}{12} \sum_{n=t-11}^{t} w_i(t)$$

Weather input variables $W_1(t)..W_7(t)$ are meteorological data gathered in Rome in 2011. Air temperatures in zones $T_1(t)..T_{10}(t)$ are provided by HAMBASE simulator.

While the simulator can use more variables, such as comfort of employees, we do not strictly need those in our experiments. [2]

4.4 Experiments

The main objective is to test QBC, hence all other aspects of the experiment are kept to be as simple and effective as possible. For that reason, the model used for prediction in these experiments is a simple linear regression model.

The output variable of the model is the gas consumption. The input variable are weather related variables, air temperatures and set point. Among those, only the temperature setpoints are controllable and the task of our active learning is to excite those inputs efficiently and save some effort and time needed to acquire training data sufficient for building a good predictive model.

Due to the absence of an initial set, we create one using random sampling strategy. This set is used through all experiments, minimizing the experiment random characteristic. The size of this initial set has been set at 10 instances, which would translate into one workweek of measuring before the experiment.

The size of committee has been set to five. Even though using less than five results in a shorter computation time, prediction itself was not as effective. However, using more than five resulted in the increase of computation time with no substantial increase in efficiency.

An issue that was more complicated was found in optimization when searching for maximum disagreement of members of the committee. Unlike in the previous demo, we now deal with multiple variables. The need of constricting variables might furthermore complicate things. In the end, we settled for a swarm based MATLAB algorithm to find the maximum standard deviation of committee. constricting was done in two ways. The first input space for synthesis of the data had similar range as the random querying strategy, the second had a wider range to see how a more vague range of the input space affects the process.

As we found out in our demo experiment before, using linear regression models as a members of committee in QBC does not work. The first experiment therefore goes only a little bit further, and uses quadratic regression model, which is chosen especially for its low computational requirements and high efficiency.

Next models selected were regression trees and neural networks. The regression trees were selected for their ability to quite easily work with multiple variables, although pruning is required for them to be the most efficient. We only used the trees in their non-pruned form because pruning 5 trees every iteration took an extensive amount of time (even longer than training 5 neural networks). Neural networks were kept as simple as possible while retaining most of their accuracy to shorten the simulation time.

4.5 Results

Efficiency of various models used as members of committee in QBC was measured by a Mean Absolute Percentage Error (MAPE):

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|$$

where y_i is real value of an instance x_i, $\hat{y}_i$ is estimated value of x_i and n is total amount of $[x, y]$ pairs. Real values were obtained from randomly sampled instances x via simulation HAMBASE, with weather variables taken from meteorology station in Ciampio, Italy, in 2011.

As in the previous chapter, our evaluation criteria are derived from MAPE values. AUTC criteria represents the sum of MAPE over all iterations and the number of learning iterations (steps) before MAPE reaches and stays below certain threshold. These criteria are later compared between the two methods (QBC with different models used for committee).

Threshold is set the same for all models. Method used to get the value of the threshold is again similar as in previous experiment. After obtaining data from random sampling strategy, the threshold is decided as a value where the MAPE has low variance.

	Polynomial	Tree	Neural Network	Random
AUTC	19.902	18.2823	21.3851	18.986
Step count	30	28	18	28
Savings	107%	100%	64%	–

Table 4.2: Results of average performance with different regression models.

Values in 4.2 are values averaged across at least ten runs with given models substituting committee members. AUTC values are generally higher than they would be in practice, as they are computed across the whole run of 68 days. Generally, we do not need that much time to acquire satisfying results.

The row with savings is our main concern in this table. AUTC is analyzed later. Savings represent how efficient the given models are in comparison with random sampling strategy. Although our main concern lies with variables $S_A(t)$ and $S_W(t)$, regression models are still created with all 19 input variables, which poses challenge for early accuracy using certain models, especially with polynomial models.

While using polynomial regression models and regression trees as members of committee did not yield satisfying results, neural networks models managed to raise the efficiency of QBC above the random sampling strategy, thus fulfilling the expectations for this experiment.

Following on the next page, all results of different models are discussed in more detail.

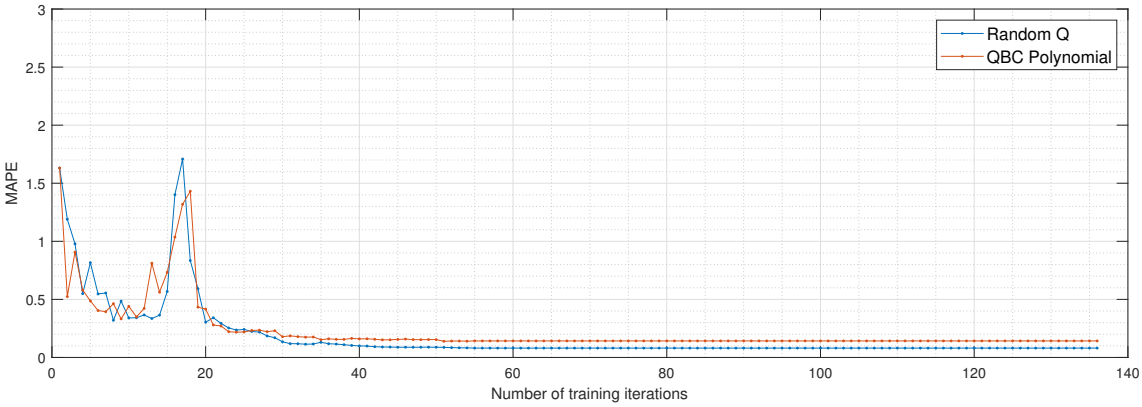


Figure 4.b: Averaged AUTC of QBC with polynomial models

First experiment was using quadratic polynomial regression for the models in the committee. Multiple different variations of polynomial models were experimented with. Regrettably, with 19 input variables, all of them could only create polynomial models of second degree, which are not optimal for QBC (similar to linear model problem discussed in 3.4).

Results (as shown in fig. 4.b) tell us that quadratic polynomial regression does not satisfy our needs for a model. Both QBC and random querying have similar performance, although QBC relies heavily on minimizing method. During initial steps, all models created for committee have very low variance, resulting in minimizing method being almost random in character (or choosing marginal points if able to).

When sufficient number of points for learning set is gathered, polynomial models quickly catches up to other used methods. In this case, the learning set has to have at least 19 points in order to satisfy polynomial models. In case of solving a problem with fewer number of inputs, polynomial models might perform better.

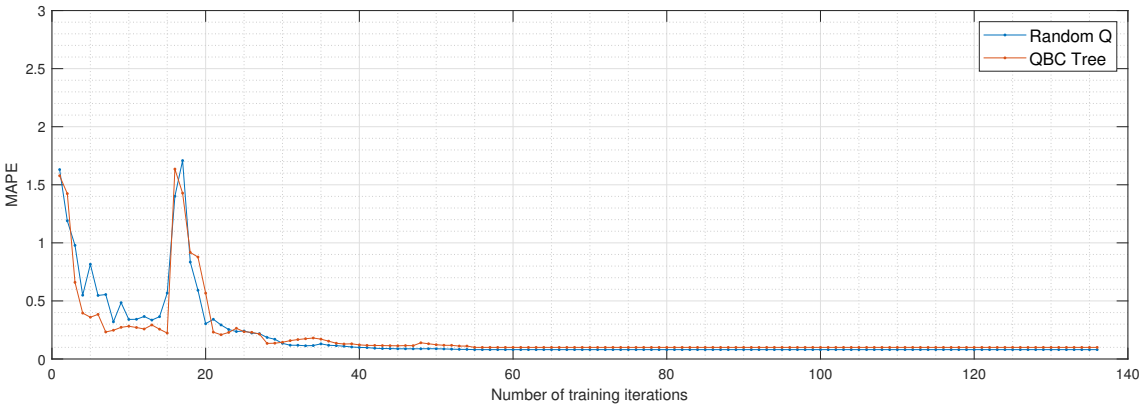


Figure 4.c: Averaged AUTC of QBC with regression tree models

Second experiment was using tree regression for the models in the committee, with averaged AUTC shown in fig. 4.c. At first, it might not seem to be much different from the results obtained with models being polynomial, but trees are actually more consistent between multiple runs.

Tree models were used right after initial regression was done. Pruning or other methods of optimizing regression trees were not used in resulting experiment for their time complexity with relatively small gains in accuracy.

Despite the fact that trees only managed to be as effective as random sampling strategy while taking significantly more time to do so, better results might be achieved with slightly lighter conditions for the models, especially in its dimension.

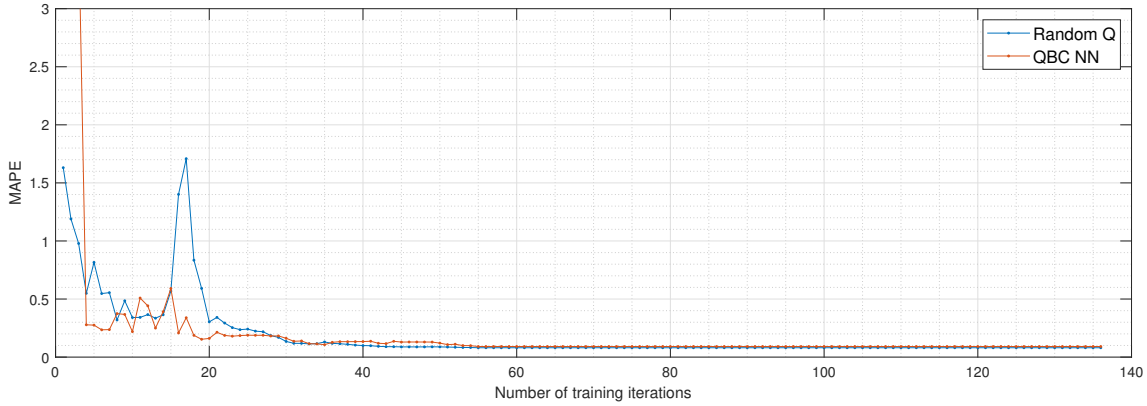


Figure 4.d: Averaged AUTC of QBC with neural network models

The final experiment was using neural network regression for the models in the committee. As opposite to previous two models used, NNs had large MAPE at the beginning. Despite the MAPE being large at the beginning, it quickly decreases and actually does not even fluctuate as much as all other methods (as can be seen in fig. 4.d).

NNs were created with 3 hidden layers. Layers were small in size, reaching maximum of four neurons per layer. With these settings, NNs managed to outperform random sampling strategy and successfully conclude this experiment.

With average 64% steps opposite to random strategy needed to reach the threshold, NNs need less than 20 iterations to find the optimal training set for prediction. Less than 20 iterations translates to just under 10 days of the system learning and that would turn into just two workweeks to fine-tune the heating plant. Of course, that is just hypothetical, as we would have to assume that the system could alter the temperature in the office building as it wished during those two weeks, which will not be always possible.

4.6 Discussion

QBC using NNs models as its committee has shown better efficiency, speed and steadiness. Although we managed to receive such results only with one type of regression models, it is not completely unexpected. Let us discuss why in this section.

First and foremost, this task can not be compared with the training task set up in chapter 3. Polynomials are easily approximated using regression in two dimensional settings. The size of initial training set does have significant importance on the overall performance of the method, because we only need as much initial points as the degree of the polynomial model is. It can even work with lower polynomial degrees at the beginning, raising the polynomial degrees later with the increased learning set size.

In this task, polynomial models are not unusable, they just need a little bit more favourable conditions to be comparable to other regression models. The largest obstacle for them is a large number input variables, in this case it being 19. With one output, it still makes it twenty dimensional problem - a very complex one to solve. For QBC to use polynomials effectively, it needs to be able to create polynomial of degrees higher than two, which was not possible with all regression methods tried.

Regression trees and NNs did not fare bad in comparison to polynomial regression. However, the task was to compare them to random sampling strategy, where they encountered strong opponent.

Regression trees only managed to perform similarly as the random sampling strategy, although they were more consistent in they learning curve. Their main pros were their speed (compared to NNs) and simplicity in setting up. Their con is especially their inability to work with small initial data set. Tree models mostly queried marginal points of examined interval and were not able to change it until external interference made them to.

NNs yielded the best results. Not only did they outperform all other regression models used for committee in QBC (which was expected), they outperformed even the random sampling strategy. Although they had the best results, they required the most computing time. Training five NNs each iteration is a costly process.

To sum up, deciding what method is the best is not only matter of the result of this experiment. There are many other factors influencing the results of similar experiments that depend on the task, such as the cost of querying, the time between querying. Nevertheless, there are also influences from this simulation, such as acquiring the test set, insufficient quality of regression models or in some cases, being lucky.

In the case of this experiment being applied in practice on the real heating plant, the fact that NNs take two minutes and random sampling is done in a few seconds does not have

such and influence as the number of queries needed to be taken. If it is possible to have enough samples gathered in two weeks, then method that would require one month to do the same is much less valuable.

Chapter 5

Conclusion

The aim of this thesis was to continue and enhance training efficiency of the prediction model for predictive control of a heating plant from [2]. We have proposed an active learning strategy that can be used for construction of a training set for prediction of continuous variables and we have used that strategy in our conducted experiment. The strategy is based on query by committee that was inspired by membership query synthesis.

We have implemented, tested and analyzed the proposed strategy on a curve-fitting task, in order to test whether the strategy can even be used. Initial results were promising, with enhancement up to 50 % of fitting model to a given curve. This experiment confirmed our strategy, even though the task has been simplified.

As the main focus of the thesis, we have used the proposed strategy on the time-series forecasting task and compared it to the strategy used in [2]. Unfortunately, during the forecasting task, a few obstacles came to the light. Committee members advanced in complexity in the form of increase of variable amount and a nonlinear character of models. Results were mostly in favour of the originally used random querying and QBC only managed to be somewhat more stable, until NNs were used for models in committee. NNs had managed to outperform the strategy, only needing about 60% time that the [2] strategy took to achieve similar results. Discussion about such results took place, trying to give insights into why were these results encountered.

Bibliography

[1] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee", in *Proceedings of the fifth annual workshop on Computational learning theory*, ACM, 1992, pp. 287–294.

[2] M. Macas, F. Moretti, A. Fonti, A. Giantomassi, G. Comodi, M. Annunziato, S. Pizzuti, and A. Capra, "The role of data sample size and dimensionality in neural network based forecasting of building heating related variables", *Energy and Buildings*, vol. 111, pp. 299–310, 2016.

[3] B. Settles, "Active learning literature survey", University of Wisconsin-Madison Department of Computer Sciences, Tech. Rep., 2009.

[4] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active learning with statistical models", *Journal of artificial intelligence research*, vol. 4, pp. 129–145, 1996.

[5] S. Caron, T. Heskes, S. Otten, and B. Stienen, "Constraining the Parameters of High-Dimensional Models with Active Learning", *arXiv preprint arXiv:1905.08628*, 2019.

[6] J. Yao, Y. Wu, and H. Zhai, "Speeding up Quantum Few-Body Calculation with Active Learning", *arXiv preprint arXiv:1904.10692*, 2019.

[7] N. Khoshnevis and R. Taborda, "Application of pool-based active learning in physics-based earthquake ground-motion simulation", *Seismological Research Letters*, vol. 90, no. 2A, pp. 614–622, 2019.

[8] S. Yu, X. Luo, Z. He, J. Yan, K. Lv, and D. Shi, "An Improved Sampling Strategy for QBC Algorithm and its Application on Gas Sensor Array Signal Processing", in *2018 Ninth International Conference on Intelligent Control and Information Processing (ICICIP)*, IEEE, 2018, pp. 224–228.

[9] I. Hossain, A. Khosravi, I. Hettiarachchi, and S. Nahavandi, "Batch Mode Query by Committee for Motor Imagery-Based BCI", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 1, pp. 13–21, 2018.

[10] R. Gilad-Bachrach, A. Navot, and N. Tishby, "Query by committee made real", in *Advances in neural information processing systems*, 2006, pp. 443–450.

[11] A. K. McCallumzy and K. Nigamy, "Employing EM and pool-based active learning for text classification", in *Proc. International Conference on Machine Learning (ICML)*, Citeseer, 1998, pp. 359–367.

[12] E. Alpaydin, *Introduction to Machine Learning*, ser. Adaptive Computation and Machine Learning series. MIT Press, 2014, ISBN: 9780262325752. [Online]. Available: https://books.google.cz/books?id=7f5bBAAAQBAJ.

[13] F. Olsson, "A literature survey of active machine learning in the context of natural language processing", 2009.

[14] I. The MathWorks. (2019). MATLAB Optimization Toolbox. The MathWorks, Natick, MA, USA, [Online]. Available: https://uk.mathworks.com/help/optim/index.html (visited on 05/12/2019).

[15] I. Guyon, G. C. Cawley, G. Dror, and V. Lemaire, "Results of the active learning challenge", in *Active Learning and Experimental Design workshop In conjunction with AISTATS 2010*, 2011, pp. 19–45.

[16] M. A. Henson and D. E. Seborg, *Nonlinear process control*. Prentice Hall PTR Upper Saddle River, New Jersey, 1997, pp. 233–309.

[17] I. The MathWorks. (2019). MATLAB Simulink. The MathWorks, Natick, MA, USA, [Online]. Available: https://www.mathworks.com/products/simulink.html (visited on 05/21/2019).

[18] M. De Wit, *Hambase: heat, air and moisture model for building and systems evaluation*. Technische Universiteit Eindhoven, 2006.

[19] A. Van Schijndel, "HAMLab: Integrated heat air and moisture modeling and simulation (Ph. D. thesis)", *Technische Universiteit, Eindhoven*, 2007.