

From Limited Annotated Raw Material Data to Quality Production Data: A Case Study in the Milk Industry (Technical Report)

Don't Cry Over Spilt Milk

Roe Shraga
shrage89@campus.technion.ac.il
Technion, Haifa, Israel

Gil Katz
gil@afimilk.co.il
Afimilk, Afikim, Israel

Yael Badian, Nitay Calderon,
Avigdor Gal
{yaelbadian,nitay}@campus.technion.ac.il
avigal@technion.ac.il
Technion, Haifa, Israel

ABSTRACT

Industry 4.0 offers opportunities to combine multiple sensor data sources using IoT technologies for better utilization of raw material in production lines. A common belief that data is readily available (the big data phenomenon), is oftentimes challenged by the need to effectively acquire quality data under severe constraints. In this paper we propose a design methodology, using active learning to enhance learning capabilities, for building a model of production outcome using a constrained amount of raw material training data. The proposed methodology extends existing active learning methods to effectively solve regression-based learning problems and may serve settings where data acquisition requires excessive resources in the physical world. We further suggest a set of qualitative measures to analyze learners performance. The proposed methodology is demonstrated using an actual application in the milk industry, where milk is gathered from multiple small milk farms and brought to a dairy production plant to be processed into cottage cheese.

ACM Reference Format:

Roe Shraga, Gil Katz, and Yael Badian, Nitay Calderon, Avigdor Gal. 2021. From Limited Annotated Raw Material Data to Quality Production Data: A Case Study in the Milk Industry (Technical Report): Don't Cry Over Spilt Milk. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 13 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 INTRODUCTION

The evolution of data acquisition, management, analytics, and visualization has led to coining the term *big data*. Big data encompasses emerging technological advancements such as IoT – Internet of things (acquisition), cloud computing (management), and machine learning (analytics), and providing an exciting arena for new and challenging research agenda. Big data is commonly characterized via a set of “V”s, out of which *volume* characterizes the need to gather, manage, and analyze massive amounts of data. A major

source of data is the one stemming from sensors, driven by IoT technology that offers standards for bringing multiple sources of data together to be stored, analyzed, and visualized.

Industry 4.0 is as an encompassing term to sensor data deployment using IoT technologies in factories where machines augmented with wireless connectivity and sensors are connected to monitor an entire production line for autonomous decision making.

Industry 4.0 faces challenges that involve utilizing big data technologies and analytics at scale. To revolutionize supply chain management, there is a need to handle the accelerating scale, scope and depth of data. The end goal is to enable complex supplier networks to optimize the flow of products, information, and finances. Such optimization allows companies to improve their overall business efficiency, while hand-in-hand personalize their services and tailoring them to customer needs, thus creating sustainable value.

Modern supply chain companies run sophisticated workflows (manufacturing, transportation, sales, logistics, etc.) and make use of IoT technology to receive streams of data that flood in from suppliers, customers, machinery, environment, and delivery channels. For example, sensor-equipped machines offer raw material readings that may be used for anticipating the performance of raw material in a production line. One of the toughest contemporary obstacles involve the need to integrate multiple data sources on a massive scale to capitalize on this rich pool of data and create invaluable knowledge and insights [61] (big data *variety*).

In this work we tackle a problem of creating a training dataset for controlled learning of the impact of raw material on a production line under severe limitations of data acquisition. Despite the generous availability of sensor data both on the side of the raw materials and on the side of the production line, it is still a challenge to construct a training dataset from existing data mainly due to the integration problem. While the raw material may be monitored carefully, once it is being shipped to the factory it may be combined with other raw material and then spread out to serve in multiple production lines, all of which may lack any control and monitoring. To allow a controlled setting, the raw material provider and the manufacturer need to agree on a protocol of providing raw material of a certain quality to be processed in a specific production line, so as to keep as many of the variables that may affect the production line in a controlled state.

We propose an active learning solution to effectively acquire IoT data for quality training purposes. Our contribution is threefold:

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. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Conference'17, July 2017, Washington, DC, USA

© 2021 Association for Computing Machinery.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...\$15.00

<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

- **Modeling:** We propose a model for unifying the representation of raw material's IoT sensor data in a production line, which lends itself well to (active) learning tasks.
- **Algorithmic:** We extend active learning state-of-the-art to effectively solve regression-based problems and suggest qualitative measures to analyze learners performance.
- **Application:** We provide a detailed dairy production industry case study to illustrate the potential of active learning to creating datasets that adhere to the application's need.

Using a controlled experiment, we demonstrate the effectiveness of applying active learning to generate a dataset for testing the impact of milk features on the production of cottage cheese. The rest of the paper is organized as follows. We start with an illustrative case study from the milk industry, demonstrating the need and possible solution to the creation of a training set using a small amount of sample collection (Section 2). A model for integrated sensor systems and a problem definition are given in Section 3. Section 4 offers our solution to effectively learn the impact of raw material on production outcome. An empirical evaluation (Section 5) is followed by related work (Section 6) and a concluding discussion (Section 7).

2 CASE STUDY: THE MILK INDUSTRY

In this section we provide a specific use case of the utilization of IoT in the food industry. We focus on the milk industry and start our presentation with a description of the farm setting (Section 2.1), followed by the dairy setting (Section 2.2). We conclude with a description of the available sensor and auxiliary data (Section 2.3).

2.1 Milk Farm's Perspective

Milk has played an important role in mammals evolution, as the most effective nutrient, contributing to survival success of these mammals descendants. Humans exercised lactation control in mammals, for nutritional purposes, since the beginning of civilization [7] and bovine milk is the major raw material in the dairy industry.

Cows must calve to produce milk and a lactation cycle is a period between one calving to the next. This cycle is split into four phases, namely early, mid, late lactation of actual milk production (each approximately 100 days) and a dry period (approximately 65 days). The next calving begins once the dry period is over.

To produce milk during these four stages, cows forgo physiological changes from mobilizing body reserves at the beginning of the cycle through body reserves regain and finally rumen rehabilitation before next calving [17, 26]. Dairy farmers objective is to obtain an optimal control of the lactation cycle by planning cow performance that maximizes the farm income [16].

Over the last century, the production of dairy cattle milk was increased enormously, mainly due to the development of quantitative population genetics [53]. The modern high producing dairy cows are a product of genetic selection for increased milk production as a primary trait [21, 58]. A modern milk production procedure takes into account the energy transfer from digestion of feed, tissue metabolism and muscle activity of cows [29]. A herd (group of cows) is managed using feed and reproduction strategies, as well as health control through the lactation cycle.

The basis for farmers' decision making is the energy balance of the individual cow. However, the production of a farm is an

aggregated effort of the whole herd. At any given time, the cows in the herd differ in their lactation stage, physiological condition, health status *etc.*, all of which have direct implications on the bulk milk, which is composed of all individual milked cows.

Milk production enjoys economy of scale. Therefore, a dominating trend in global dairy industry involves a drop in the number of herds along an increase in the herds sizes. Owner operated farms are evolving into big organizations [15], small family farms evolve into large dairies with thousands of animals, and traditional 10-50 cows family farms built in populated areas are replaced with large dairies in agricultural areas, with thousands of milking cows.

While intimate familiarity with each individual animal is no longer an option in large dairies, the basic production unit remains the individual cow with its lactation cycle. Therefore, monitoring a heterogeneous collection of basic production units remains a necessity in this industry, leading to the creation of *precision dairy farming* (PDF). PDFs are computerized management systems, such as afifarm 5, which was developed by our industrial partner Afimilk,¹ and uses information technology to manage the smallest production unit possible, expressing its genetic potential in accordance with the economical goal and animal well-being. Afifarm 5 uses a reliable ID system, a set of sensors to automatically collect relevant data and the farming know-how to perform analysis of the data to support productive decision making. IoT sensors provide detailed online data about each individual cow in a herd. With proper physiological labeling and interpretation, this data can be used to support the decision making in farms [46, 59].

2.2 Milk Dairy's Perspective

Milk is a biological complex fluid. While the major component of milk is water or skim liquid, milk contains up to 14% of total solids, double than most vegetables. Milk contains fat, protein, lactose, minerals, and more than 250 chemical compounds and metabolites such as micronutrients (e.g., vitamins), amino acids, trace minerals, hormones, and enzymes [4, 27]. The rich content of milk allows it to be manufactured into other, stable dairy products such as butter, cheese, dried milks, ice cream, and condensed milk using complex and sensitive processes that have been refined over many generations. To date, behavior of complex fluids like milk, whose dynamics are highly nonlinear, is still not well understood and self assembly into different types of aggregates is complex [19, 20]. Aiming to understand these dynamics per a required product is at the heart of the know-how of the dairy production line.

Consider *cheese*, a product of a formation of protein matrix originating from casein micelles held together by weak physical forces trapping fat globules within the matrix: "The physical properties of cheese (*i.e.*, body/texture, melt/stretch, and color) are influenced by initial cheese[-]milk composition, manufacturing procedures, and maturation conditions. Two of the most important factors influencing these properties are the condition of the CN [(casein)] particles in cheese (e.g., interactions between and within molecules, as well as the amount of Ca [(Calcium)] associated with these particles) and the extent of proteolysis. These are in turn influenced by various environmental conditions such as pH development, temperature,

¹Afimilk LTD, Afikim, Israel, <https://www.afimilk.com/afifarm>

and ionic strength. Therefore, how individual CN molecules, or aggregates of many CN molecules interact, is vital in understanding the physical and chemical properties of cheese.” [40].

Complex fluid dynamics led to a setting where there is no uniform set of principles to fully control milk processing. Milk processed at the production line has low repeatability between days, production lines, and farms. Moreover, milk batches containing the same concentrations of solids, fat, protein, and even casein were shown to result in different cheese yields [35]. Contemporary large scale dairy plants use inaccurate measures such as solids level (fat and protein), Somatic Cell Count, and bacteria counts (herd udder health) to quantify daily milk quality and to scale milk payment. It is this challenge we aim at tackling in this work, offering an effective algorithm for quality data acquisition using IoT data.

2.3 From Farmer to Dairy

The farmer (Section 2.1) and dairy processors (Section 2.2) address two different qualitative evaluation of milk. On the one hand, at the dairy, milk quality is measured on an industrial level and related to economics through product composition and suitability, as well as the production yield. Milk age and storage conditions influence milk quality as well [43]. Finally, end user needs regarding good and bad fatty acids profiling should also be taken into account. On the other hand, farmer’s milk quality, properties, and composition are addressed at the udder level and depend on breed, nutrition, lactation cycle, and udder infection [36, 49]. The interaction between farmer and dairy occurs in the farm’s bulk tank. The input from the farmer to the bulk tank is milk from selected individual cows (farmer’s perspective). The output of the bulk tank at the dairy is a composed milk from individual milked cows (dairy’s perspective).

This work attempts to remove the barrier between processor (dairy) and farmer, employing PDF data (Section 2.1). PDF data, collected from individual production units (cows), can be used to characterize bulk tank’s properties and, when merged with production line data, to optimize production. This process is a data management challenge, handling data from multiple sources, of various types (including categorical, quantitative, subjective, and objective data), and at different time scales. To demonstrate the rich IoT data content of this automated process, we use afifarm 5 to extract the following daily data:

- Individual cow wearable (afiactII motion) sensors that measure cow behavior including motions, activity, lying time, and lying bouts. Each cow is equipped with a single sensor.
- Milk data sensors using milk spectrometer and milk flowmeter that measure yield, flow, conductivity, milk components, and milk NIR spectra 2-5 times a day using aifilabTM milk analyzer and afimilk MPCTM milk meter. We note that these sensors are subject to frequent malfunctions.
- A walk-through scale, measuring cow weight.
- Third party data such as health report (Veterinarian), feed (Nutritionist), and farmer’s action reports. Such data suffers from low integrity and inconsistency across farms.

IoT data is enriched with data from production lines (yield, process parameters) and chemistry laboratories (coagulation properties, proteins profiling, saturated and un-saturated fatty acids, cheese and yogurt production pilots, and electronic sniffer).

In this work, we present a methodology, based on active learning, to create a schedule for experimenting with IoT data and build a prediction model of raw milk’s rennet activity during cheese production. Milk clotting activity may only be measured by the rapidity with which the enzyme clots milk under a set of specified conditions (RCT) or the Curd Firmness (CF) [9]. Such retrospective empirical evaluation of milk’s coagulation properties provide limited means to assess milk quality. Yet, there is a need for predictive mechanisms to assist dairies in understanding the quality of milk that enters a manufacturing line. As the labeling quality measurement, our measure of choice is CF, measured with an Optigraph instrument (Ysebaert, Freppillon, France) [32, 47]. CF is measured in the lab in a process that takes 60 to 90 minutes for a batch of 10 samples.

3 MODEL AND PROBLEM DEFINITION

In this section, we present a model of the data at the farm (Section 3.1 and Figure 1) and use it to introduce the problem definition that serves as the focus of this work (Section 3.2).

3.1 Model

Let $S = \langle s_1, s_2, \dots, s_n \rangle$ be a set of sensors (ordered for convenience of presentation). We use a *point-based semantics* for time and for simplicity sake we use \mathbb{Z}^+ as a time domain. Using S and a discrete time domain, the data emitted by sensor $s \in S$ is modeled as a data stream $s^s = \langle e_1^s, e_2^s, \dots \rangle$, which contains a (possibly infinite) sequence of *events*, with a similar structure and meaning. A sensor s belongs to a sensor type, denoted by $s.type$, which determines the structure of events generated by s . Such a structure of an event is given as a set of attributes, $\mathcal{A}^s = \{a_1^s, a_2^s, \dots, a_p^s\}$, with $dom(a_i^s)$ as the domain of attribute a_i^s ($1 \leq i \leq p$), dictated by its type. We denote by $dom(\mathcal{A}^s) = \times_{1 \leq i \leq p} dom(a_i^s)$ the domain of sensor s , where \times is the generalized Cartesian product. We use $e.a^s$ to denote the value of attribute a^s of event e and $e.\mathcal{A}^s$ to denote the values assigned with event e . Finally, the joint domain of all sensor values is denoted as $dom(\mathcal{A}) = \times_{s \in S} dom(\mathcal{A}^s)$.

In common to all events is an attribute $e.id$, which uniquely identifies the event. We also assume that each event e is associated with a timestamp attribute $e.ts$, referring to the time it was emitted. We require each sensor to emit $e.cid$, explicitly creating the notion of a *case* to identify a unique identity that associates all events that share the same case id. Let C be a set of *cid* values and $c \in C$ a single case id. A *sample* of S at time $t \in \mathbb{Z}^+$ for case c is defined as $x_{c,t} = \langle e_{c,t}^{s_1}.\mathcal{A}^{s_1}, e_{c,t}^{s_2}.\mathcal{A}^{s_2}, \dots, e_{c,t}^{s_n}.\mathcal{A}^{s_n} \rangle$, where $e_{c,t}$ stands for an event e such that $e.cid = c$ and $e.ts = t$.

Example 3.1. Let $S_{ex} = \langle s_{spectrometer}, s_{status} \rangle$ be an example of two sensors based on our case study (Section 2). The sensor $s_{spectrometer}$ represents the dynamics of milk coagulation process and composed of the attributes $\{fat, protein, lactose, IgG, casein, Oa, Sufa, Mufa\}$. *casein* with domain $dom(casein) = [1.5, 3]$, for example, represents κ -casein hydrolysis [44], which is the primary phase of the enzymatic milk coagulation. Other attribute examples in $s_{spectrometer}$ are *Oa*, *Sufa*, *Mufa* measuring Oleic Acid and Poly-unsaturated fatty acids and Mono-unsaturated fatty acids, respectively. *sstatus* tracks cows status with, among others, the attributes days in milk (*DIM*, $dom(DIM) = \{0, 1, \dots, 305\}$) and *GYN* (gynecological status).

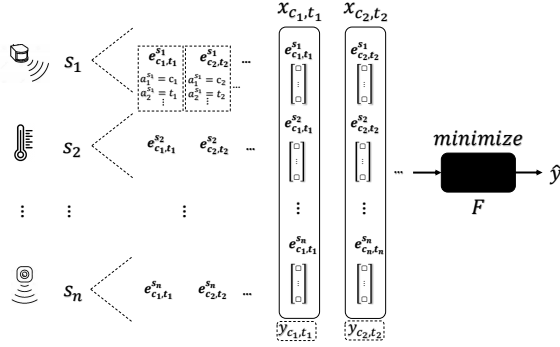


Figure 1: Farm Model Illustration

In the context of a production line, we define an outcome measure Y to be some assessment of performance over a sample, which takes its values from $\text{dom}(Y)$. Given a sample $x_{c,t}$, we mark by $y_{c,t} \in \text{dom}(Y)$ the measure assigned with it. $f : \text{dom}(\mathcal{A}) \rightarrow \text{dom}(Y)$ maps sensor values to a measure.

3.2 Problem Definition

The process of collecting milk, producing cheese, and analyzing its quality is expensive, requiring many resources. Hence, number of samples available for learning is limited. Moreover, the dairy can only allocate limited facilities for the benefit of an experiment and therefore only a few samples can be processed each week. We therefore have a constrained setting with a limited time for experimenting (an *experiment horizon* $\{1, 2, \dots, T\}$) and a limited number of samples per iteration (a budget b).

We position this problem as an *active learning* [56] problem, aiming at selecting the b most informative samples for labeling throughout the experiment horizon $\{1, 2, \dots, T\}$.

A common setting for active learning assumes the availability of a static pool from which samples are drawn. The sample collection in the milk industry presents a unique challenge, in which we do not have such a privilege. In our scenario, we are presented with an infinite domain of samples $\text{dom}(\mathcal{A})$ from which, at each time t , we are given a selection of a different set of authorized samples X_t^a . To be more concrete, each week our sample set changes based on the specific characteristics of milk that cows provide. The variation depends on external aspects (e.g., weather, food provision) as well as internal (e.g., cow health, days into milking cycle).

Our goal is to find a mapping from input samples in $\text{dom}(\mathcal{A})$ to an outcome measure $\text{dom}(Y)$, by selecting at time t the best available samples X_t out of X_t^a , subject to a budget constraint.

PROBLEM 1. Let $\{1, 2, \dots, T\}$ be an experiment horizon and b a sample budget. At time $t \leq T$, given X_t^a and knowledge of time $t-1$: $X_{t-1}, Y_{t-1}, f_{t-1}$ ($X_0 = Y_0 = \text{null}$ and accordingly $f_0(X_0) = \text{null}$) we seek a set of samples X that minimizes some function F , as follows.

$$X_t = \arg \min_{X \subseteq X_t^a: |X|=b} F(X, X_{t-1}, Y_{t-1}, f_{t-1}) \quad (1)$$

We propose several alternatives to the function F , adapting known measures for active learning to the unique characteristics

of our problem. We also test different model generators f in the incremental setting of active learning.

4 SAMPLE COLLECTION SCHEDULE

Learning from data of the milk industry is challenging. A prime reason for that is the high cost of labeled data that require expensive physical experimentation. Therefore, we aim at making the most out of each labeled sample. To do so, we introduce in this work an *active learning* approach, aiming at maximizing performance with as few labeled samples as possible [56]. With active learning, one chooses samples from which most learning can be performed.

Problem 1 requires the generation of a sample collection schedule. In this section we propose such a schedule, partitioning it into two phases, an initialization phase in which samples are scheduled to be collected according to some pre-specified rule, and an active learning phase, in which decisions are taken based on input from previous steps. In the absence of an initial training set we propose a bootstrapping approach to support an intelligent sample selection. The initialization step does not consider a trained model or labels, and uses human meta-knowledge instead, to identify useful samples. Once a sufficient amount of samples to ensure effective learning are labeled, the learning step can begin. At one extreme, the full schedule can be determined solely using the initialization phase without actual learning. At the other extreme, initialization is performed for a single instance, after which learning comes into play. In Section 5 we test for the best combination of the two phases.

Figure 2 illustrates the active learning framework we use in this work. An active learner asks queries in the form of unlabeled samples to be labeled by an oracle, which in our case is the cheese production process outcome. We iteratively construct a model based on these labels. Using (unlabeled) preprocessed data, an initial pool of samples is selected for labeling to construct a baseline model. Then, iteratively, based on some criteria, an additional set of unlabeled samples is selected for labeling. Using these labels, the model is updated and evaluated. This process is repeated until some stopping criteria is achieved (e.g., the desirable performance of the model is obtained). Altogether, the key ingredient in active learning is a selection method to choose useful samples to collect and label, complying with the given sample budget.

We first introduce several methods that serve for the initialization phase (Section 4.1). Then, we present active learning for regression problems and focus on active learning selection methods (Section 4.2). Evaluation of the sample collection schedule is discussed in Section 4.3. We conclude the section with our proposed algorithmic solution (Section 4.4).

4.1 Initialization Methods ($\text{init}(\cdot)$)

In the initialization phase we choose a set of samples using some non-learning strategy. A naïve strategy may be to randomly choose from the pool of available samples. Next, we introduce three additional strategies that are based on some human meta-knowledge. In what follows, we present the strategies under a constraint of one sample at a time, noting that since no learning is involved between sample executions, multiple samples per iteration may be selected iteratively. Recall that in our setting, each iteration t comes with

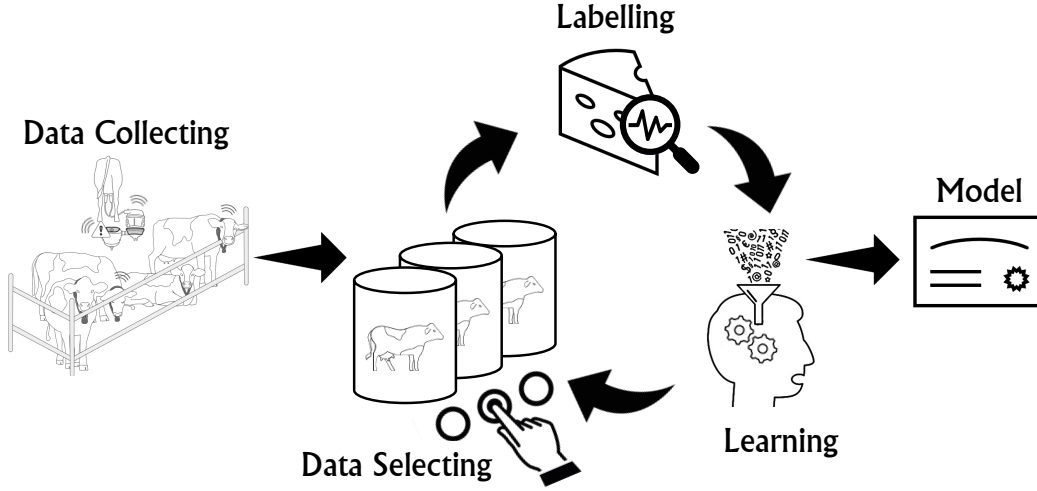


Figure 2: An Active Learning Framework

its own pool of samples X_t^a and therefore each initialization step should be handled separately.

Pareto Sampling: With domain expert assistance, we partition the feature space into two sets $\{d^+, d^-\} \subseteq \{1, 2, \dots, d\}$ such that $d^+ \cup d^- = \{1, 2, \dots, d\}$. d^+ and d^- contain features that positively and negatively affect the target value, respectively.

Using d^+ and d^- , a *pareto dominance* relation between two samples $x, x' \in X_t^a$, is given as follows. x *Pareto positively dominates* x' if the following conditions hold ($x[j]$ denotes the j th feature of sample x): (1) $\forall j \in d^+, x[j] \geq x'[j]$, (2) $\forall j \in d^-, x[j] \leq x'[j]$, and (3) $\exists j \in d^+ \mid x[j] > x'[j] \vee \exists j \in d^- \mid x[j] < x'[j]$.

Similarly, x *Pareto negatively dominates* x' if the following conditions hold: (1) $\forall j \in d^+, x[j] \leq x'[j]$, (2) $\forall j \in d^-, x[j] \geq x'[j]$, and (3) $\exists j \in d^+ \mid x[j] < x'[j] \vee \exists j \in d^- \mid x[j] > x'[j]$.

Pareto sampling selects samples from the pool of samples that are not (positively or negatively) Pareto dominated in X_t^a .

Distance Sampling (Di): A common sampling technique aims at distributing the samples across the whole feature space, under the assumption that samples that are close together add little information. Using geometric characteristics over the input features space, distance sampling (Di) selects a distant sample, one that is located (using some distance measure) as far as possible from already labeled samples (those in X_{t-1}). Formally, let $d(\cdot, \cdot)$ be a distance measure. We select a new sample $x \in X_t^a$ such that

$$\arg \max_{x \in X_t^a} \min_{x' \in X_{t-1}} d(x, x') \quad (2)$$

In the experiments we use Euclidean distance: $d(x, x') = \|x - x'\|_2$.

Clustering sampling (Cl): Using a similar assumption as with distance sampling, we avoid using two samples that are close together. With clustering sampling, a geometric structure is used to group samples into clusters. Therefore, as a first step a clustering algorithm (e.g., K-means [45]) is utilized to generate a set of clusters

over X_t^a . Then, Cl selects a representative sample from each cluster and applies Di (distance sampling, see above) to choose a sample from the set of representatives. A representative may be chosen by an expert, or simply by choosing the cluster's median value (the one closest to a cluster mean). In the empirical evaluation we use the K-means algorithm with $K = 20$.

4.2 Active Learning for Regressors

Unique to our approach is the use of active learning for regression problems. Most methods in the literature address classification problems [11], assuming all incorrect labels equally err. In what follows, we now explain how to adopt existing query strategies, which were originally designed for classification problems to offer solutions to regression problems. While the initialization methods described above (Section 4.1) can continue and serve as the selection methodology, in this section we assume that we have at our disposal a set of already labeled samples X_{t-1} , Y_{t-1} and a respectively trained model f_{t-1} when selecting samples for $X_t \subseteq X_t^a$.

We first suggest a simple extension of Di (Eq. 2), termed **Predictions (Pr)**, utilizing a trained model. Pr selects samples based on the distance of their predicted value, using the trained model, from other known labels. Formally, let $d(\cdot, \cdot)$ be a distance measure. We select a new sample $x \in X_t^a$ such that

$$\arg \max_{x \in X_t^a} \min_{y' \in Y_{t-1}} d(f_{t-1}(x), y'). \quad (3)$$

In the experiments we use absolute distance: $d(f_{t-1}(x), y') = |f_{t-1}(x) - y'|$. We divide the remaining proposed methodologies into three groups, namely, Uncertainty, QBC, and EMCM, as follows:

Uncertainty: This selection method selects a sample at time t that the current model is least certain about its label. For classification tasks, it is usually a straightforward approach, selecting a sample with the lowest margin between predicted classes, or a sample with

the highest classification entropy. Such methods use the probability of a classifier to measure classification uncertainty. Switching from classification problems, where labels are discrete, to regression with labels from a continuous domain, the probability of obtaining a specific label is zero. Therefore, for a regression model, uncertainty is assessed, for example, by selecting a sample with the highest prediction variance. Note that in a regression scenario, selecting outliers may considerably change the model (even worsen performance), while for classification, it has only a minor effect on the decision boundaries [50]. Next, we propose three approaches to compute uncertainty for regression models:

Uncertainty Discretization (UDi) discretizes the regression label, converting the label space (\mathbb{R}) into classes (\mathbb{N} , as in classification). Applying a discretization can utilize expert knowledge to determine a transformation from \mathbb{R} to \mathbb{N} . Alternatively, the transformation can be done heuristically, dividing the output space into equal (weighted) bins. Then, we can transform Y_{t-1} to classes and train a model f'_{t-1} over X_{t-1} . Once we have a trained f'_{t-1} , it is used to compute the uncertainty using, e.g., classification entropy [56].

Uncertainty Clustering (UCI) uses clustering to estimate the uncertainty of a regression model. UCI selects low quality representative samples by first applying a clustering algorithm to generate a set of clusters over X_t^a (in the experiments we used the K-means algorithm with $K = 20$). Then, UCI chooses k clusters having the highest variance by applying f_{t-1} for each sample in a cluster. Finally, UCI uses the silhouette measure [54] to select samples with the lowest coefficient out of the chosen k clusters.

Uncertainty Mean Squared Error (UMSE) is a novel approach we propose to measure regression uncertainty for a sample x using a weighted mean squared error as follows:

$$UMSE(x) = \frac{\sum_{(x', y') \in X_{t-1}} \frac{1}{d(x, x')} (y' - f_{t-1}(x'))^2}{\sum_{(x', y') \in X_{t-1}} \frac{1}{d(x, x')}} \quad (4)$$

where $d(\cdot, \cdot)$ is a distance measure (Euclidean distance). The nominator uses inverse distances to weight the model squared errors, where larger error corresponds to higher uncertainty. The denominator is a normalization factor. Intuitively, a sample that is close to labeled samples with large squared error, has a higher uncertainty. A sample with the highest UMSE score is selected.

Query-by-Committee (QBC) generates a committee of models from a labeled training set. A committee may include multiple trained models on different subsets of the labeled data or variations of the same model with different hyperparameter settings. A possible committee may also include models that use different algorithms (e.g., linear regression or random forest).

Once a committee of models is generated, the selection is based on the committee disagreement. For classification tasks, disagreement can be measured by voting entropy, selecting samples for which the entropy of the votes among the committee is the largest [14]. For regression tasks, the disagreement can be measured by the sum of squared error (or variance) within the predictions of the sample [10]. This, in turn, quantifies a finer computation of the disagreement among a committee of regressors.

Formally, let $f_{t-1}^1, \dots, f_{t-1}^C$ be a set of C trained models over X_{t-1} . The disagreement of the sample x is $\frac{1}{C} \sum_{c=1}^C (f_{t-1}^c(x) - \bar{y}^c)^2$, where \bar{y}^c is the predictions mean of the c 'th model.

Expected Model Change Maximization (EMCM) selects samples at time t whose labels are expected to change the model the most at that time [11]. A model change depends on the algorithm of choice. For example, when using linear regression, model change can be expressed as the difference between the regression line coefficients of the current model and those of the updated model. It is worth noting that such a selection is usually aimed at a faster convergence. Given that the true target value of the candidate sample is unknown, we generate a committee of models to estimate the model change. We note here that whereas QBC selects samples based on the level of disagreement within a committee, EMCM selects samples based on the level of disagreement between the committee and the actual model.

In this work we follow a method suggested by Cai *et al.* [11]. Let f_{t-1} be a model with parameters θ that minimizes some training loss and θ' be the parameters of a model f'_{t-1} that minimizes the loss by adding the sample (x, y) to X_{t-1} . Then, the model change is given by $\|\theta - \theta'\|_2^2$, measuring the difference between the model parameters with and without the sample x . To set the proper set of parameters θ' , we use stochastic gradient descent [8], according to which, we update the model parameters θ following a negative trend of the training loss. Specifically, if f_{t-1} is a linear regression model, θ is the coefficients and its loss is given by MSE (see Section 4.3). The MSE of the sample (x, y) is $(\theta^T x - y)^2$ and using SGD we obtain $\theta' = \theta - 2\mu x(\theta^T x - y)$ where μ is some learning rate. Accordingly, The model change is $\|2\mu x(\theta^T x - y)\|_2^2$. We estimate the true model change using a committee of C models, as follows: $\frac{1}{C} \sum_{c=1}^C \|2\mu x(\theta^T x - f_{t-1}^c(x))\|_2^2$ and select that sample that maximizes it.

4.3 Active Learning Evaluation

Accompanied by the variety of active learning methods, there is a need to evaluate the learning process and compare the performance of the various methods. Shen *et al.* define a *useful sample* using three qualitative criteria, namely, informativeness, representativeness, and diversity [57, 60]. *Informativeness* dictates that selected samples should contain rich information, to allow the model to significantly benefit from labeling them. *Representativeness* suggests selecting samples that represent a large portion of the input data [25] and *diversity* states that samples should scatter across the full input space. Overall, representativeness and diversity ensure coverage of a large portion of the input space and good-sized trained data call for informativeness to enrich the model and certify its validity.

Table 1 provides a qualitative comparison between the different methodologies. For each method we specify whether it utilizes f_{t-1} , X_{t-1} , Y_{t-1} , whether it uses expert knowledge, with which qualitative criteria [57] it complies, and an approximate computation time, with S corresponding to slow, M to moderate, and F to fast.

Pareto requires no prior knowledge of the domain as determined by f_{t-1} , X_{t-1} , Y_{t-1} and only heuristically defines a ranking among samples. UMSE and EMCM use all possible resources when selecting samples. Most methods can benefit from expert knowledge, except QBC and EMCM, which rely on multiple trained models. All selection methods comply with informativeness as they utilize f_{t-1} ,

Table 1: Comparison between active learning methods. For each method we specify whether it uses a prior trained model (f_{t-1}), an existing training set (X_{t-1}) and the labels of an existing training set (Y_{t-1}). We also state whether the method requires an expert knowledge, complies with one of three qualitative criteria [57], and subjectively assess its computation time.

		Pareto	Di	Cl	Pr	UDi	UCI	UMSE	QBC	EMCM
Usage	Uses f_{t-1}	✗	✗	✗	✓	✗	✓	✓	✗	✓
	Uses X_{t-1}	✗	✓	✓	✗	✓	✗	✓	✓	✓
	Uses Y_{t-1}	✗	✗	✗	✓	✓	✗	✓	✓	✓
	Uses Expert Knowledge	✓	✗	✗	✗	✓	✗	✗	✗	✗
Evaluation	Diversity	✓	✓	✓	✓	✗	✗	✗	✗	✗
	Representativeness	✗	✗	✓	✗	✗	✗	✗	✗	✗
	Informativeness	✗	✗	✗	✓	✓	✓	✓	✓	✓
Performance	Computation Time	F	F	F	F	M	M	F	S	S

Y_{t-1} or both. Cl adheres to the representativeness as it chooses a representative among each cluster. Initialization methods as well as Pr choose diverse samples using the space of samples as a guiding principle in selecting samples. Finally, considering weekly cycles of learning (as in our case study) the importance of computation time is negligible. Nevertheless, we note that while initialization methods are comparatively fast, committee-based methods (QBC and EMCM) are quite slow as they require training multiple models.

4.3.1 Qualitative Evaluation Measures. The standard evaluation measure for regressors is mean squared error (MSE) [56], or one of its derivatives, e.g., RMSE. Let Y and \hat{Y} be a set of outcomes and a set of predicted outcomes, respectively. MSE is given by:

$$MSE(Y, \hat{Y}) = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (5)$$

MSE evaluates a set of samples jointly, regardless of the evaluation timing of samples. To better understand the quality of active learning as a process of sample selection, we suggest next measures that are more suited to quantify temporal cumulative quality by assessing aggregated performance, convergence, and smoothness.

We use Area Under Curve (AUC) to measure the MSE behavior over the learning course and evaluate the performance of active learning methods over time [5]. AUC is used for classification problems but can also be applied in regression tasks. AUC is a measure of error, so the lower the AUC, the better the method. Yet, since MSE is not bounded for regression, the AUC is extremely exposed to outliers. To overcome this deficiency, we propose the use of the log of AUC (logAUC). Let $MSE_t(Y, \hat{Y})$ be the MSE of Y and \hat{Y} at the t -th timestamp. AUC and logAUC are defined as follows:

$$\begin{aligned} AUC(Y, \hat{Y}) &= \sum_{t=1}^T MSE_t(Y, \hat{Y}) \\ \log AUC(Y, \hat{Y}) &= \sum_{t=1}^T \log(MSE_t(Y, \hat{Y}) + 1) \end{aligned} \quad (6)$$

To measure smoothness, we adapt the Absolute Second Difference (ASD) measure [34], which uses second differences to capture the inflexion of the curve [6]. ASD uses $MSE_t(Y, \hat{Y})$ as follows:

$$ASD(Y, \hat{Y}) = \frac{1}{T-1} \sum_{t=2}^{T-1} |MSE_{t+1}(Y, \hat{Y}) - 2MSE_t(Y, \hat{Y}) + MSE_{t-1}(Y, \hat{Y})| \quad (7)$$

Early stages of learning may demonstrate unstable behavior as they only rely on a small set of samples. To compensate for such instability, we also use a weighted version of ASD, as follows:

$$WASD(Y, \hat{Y}) = \frac{2}{(T-1)(T-2)} \sum_{t=2}^{T-1} t \cdot |MSE_{t+1}(Y, \hat{Y}) - 2MSE_t(Y, \hat{Y}) + MSE_{t-1}(Y, \hat{Y})| \quad (8)$$

Finally, to measure convergence, we use a First to Converge (FTC) measure (inspired by the definition of a sequence limit), quantifying the extent of which the learning converges to a small difference ($\varepsilon > 0$) between consecutive iterations. Using $MSE_t(Y, \hat{Y})$, let ε be a small positive number. FTC is defined as follows:

$$FTC(Y, \hat{Y}) = \underset{1 \leq t \leq n}{\operatorname{argmin}} \{t : |MSE_t(Y, \hat{Y}) - MSE_{t-1}(Y, \hat{Y})| < \varepsilon, \forall t < i \leq n\} \quad (9)$$

We note that in the experiments we use $\varepsilon = 0.01$.

4.4 Algorithm

Algorithm 1 details our methodology, adopting active learning, of training using data that is gathered from IoT sensors. According to the setting of our case study (Section 2.2), our aim is to select samples for learning on a weekly basis. The input of the algorithm is a sequence of available sample sets over an experiment horizon $X_1^a, X_2^a, \dots, X_T^a$ and an hyperparameter $K \leq T$ representing the number of initialization steps. First we apply one of the initialization methodologies (see Section 4.1) for K steps (lines 4-7), based on which we train an initial model (line 8). We analyze the implementation of *init*(\cdot) in Section 5.2.2, choosing a methodology as well as the size of K .

Once an initial model is obtained, we utilize advanced active learning methodologies, which require an already trained model (repeating lines 9-13 for $T-K$ steps). Specifically, each round begins with selecting a set of samples for labeling $X_t^a \subseteq X_t$ (lines 10-11) using a selection methodology *select*(\cdot) (out of the ones presented in Section 4.2). Then, the new labeled samples are added to the

Algorithm 1 Data Collection Schedule Algorithm

```

1: Input:  $X_1^a, X_2^a, \dots, X_T^a, K < T$ 
2: Output: A trained model  $f$ 
3:  $X_0 = Y_0 = \text{null}$ 
4: for  $t = 1$  to  $K$  do ▷ Initialization.
5:    $X_t \leftarrow \text{select}(X_t^a)$ 
6:    $Y_t \leftarrow \text{Obtain labels for } X_t$ 
7:    $X_t = X_t \cup X_{t-1}; Y_t = Y_t \cup Y_{t-1}$ 
8: Train a model  $f_K$  using  $X_K, Y_K$  ▷ Constructing an initial model.
9: for  $t = K + 1$  to  $T$  do ▷ Active learning.
10:   $X_t \leftarrow \text{select}(X_t^a)$ 
11:   $Y_t \leftarrow \text{Obtain labels for } X_t$ 
12:   $X_t = X_t \cup X_{t-1}; Y_t = Y_t \cup Y_{t-1}$ 
13:  Train a model  $f_t$  using  $X_t, Y_t$ 
14: Return:  $f = f_T$ 

```

accumulated labeled sample set X_{t-1} (line 12), which is then used to train a new model (line 13). Finally, after a stopping criterion is reached, we return the current model as the output (line 14).

5 EMPIRICAL EVALUATION

We conducted a thorough empirical evaluation and report on its results herein. We start with a description of the experimental setup (Section 5.1), followed by report of results (Section 5.2).

5.1 Experimental Setup

We begin by describing our experimental design with an overarching goal of understanding the impact of milk features on the production of cottage cheese. We introduce the dataset based on our case-study (Section 2) and detail the experiments methodology. We further provide the code and a sample of the data in a repository [1].

5.1.1 Dataset. The dataset is composed of two daily reports from the milk farm, one characterizing the cows and the other the milk itself. Each sample represents a cow at a date, such that multiple cows data it available at a single date and different instances of the same cow are available at different dates.

Features of cows include age, number of lactations, diseases, and an activity report including number of strides and recumbence. Milk features include quantity, milking time, milking rate, and chemical content including protein, fat, lactose, somatic cells, and urea. Most features are numeric with a few exceptions of binary values. Missing values were imputed using historical data.

The proposed algorithm was implemented in a real-world experiment, jointly with a large dairy in Israel. Twenty-four Israeli Holstein Herds at different sizes varying from 80 to , 1000 individual cows participate in the study. The total daily production of the herds is approximately 400,000 Kg. The data of the individual cows comprising each herd is aggregated to herd data presentation. Once a week, our data scheduler (Algorithm 1) selects herds (samples) with production amounting to $\sim 100,000$ Kg daily. This milk is collected by the dairy and channeled to produce cottage cheese. Commercial Protein efficiency and Cheese Yield are the labels (targets) in this setting. Due to third-party confidentiality restrictions

on dairy performance measures, we used CF as a label, calculated at Afimilk laboratory (see Section 2.2), and used individual cows as samples for evaluating the algorithm various settings.

5.1.2 Methodology. Afimilk weekly reports, containing samples of 500 cows ($\forall t \mid |X_t^a| = 500$), were used to evaluate Algorithm 1. The labeling budget for each iteration is 8 cows ($\forall t \mid |X_t| = 8$). We experimented with $T = 100$ iterations, from which we allocated a maximum of 25 iterations for initialization (we validated the span $K \in \{5, 10, 15, 20, 25\}$). In addition, for validation purposes, we use an **unseen** test set to test the performance of the different methods. We analyze the following aspects of Algorithm 1:

Regression method (Section 5.2.1), examining linear [55], Random Forest [39], XGBoost [12], and Multi-layer Perceptron neural network [22] regressors.

Initialization method (Section 5.2.2), optimizing the number of initialization steps and the methods from Section 4.1.

Selection method (Section 5.2.3), testing the methods described in Section 4.2.

Active learning quality (Section 5.2.4), using the measures described in Section 4.3.

Features importance analysis (Section 5.2.5), using SHAP [41].

We tested two types of committees for QBC and EMCM, namely a bootstrap committee (QBC_{boot} and EMCM_{boot}) and a models committee (QBC_{model} and EMCM_{model}). For the latter, we use Ridge, Lasso, Linear regression, Random Forest, Gradient Boosting Machine, and KNN regressor models.

For linear regression, random forest, and multi-layer perceptron implementation we use scikit-learn² and for XGBoost we use a Python implementation³ with default parameter settings. To allow a fair comparison, we fixed the initial seed. To avoid a “cherry picking” effect, we ran the naïve strategy of random selection 15 times and report on the average performance.

Following Baram *et al.* [5], we use the logAUC to compare the performance of learning among the methods. Recall that the reported results are based on an unseen test set. In order to fairly compare and select among the different methods, we checked statistical significance with a paired two-tailed t-test to compare the $\log(MSE+1)$ values along the experiment horizon (with $p\text{-value} < .05$). When comparing multiple methods, we also add a Bonferroni correction. We have computed all significance tests comparing among methods and report on the most dominant ones throughout this section. Table 3 presents statistical significance differences of performance in terms of logAUC over random sampling.

5.2 Results

Active learning is the methodology of choice for this work, due to the scarce availability of labeled data. The goal of our research is to predict milk-cheese quality and understand the effect of raw material on production outcome. In sections 5.2.1-5.2.3 we analyze the parameters of the best active learning method to achieve this goal, testing type of regression (Section 5.2.1), initialization method (Section 5.2.2) and selection method (Section 5.2.3). Then, in Section 5.2.4 we analyze the quality of learning and finally, in

²<https://scikit-learn.org/stable/>

³https://xgboost.readthedocs.io/en/latest/python/python_intro.html

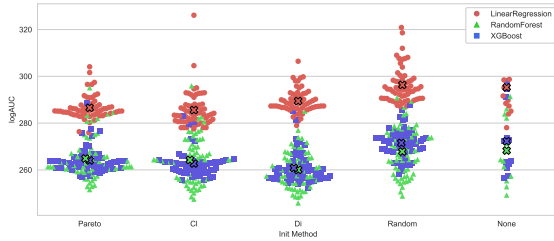


Figure 3: Comparison between the different regressors, partitioned based on initialization methods (x-axis) and evaluated using logAUC score (y-axis). The “X” denotes an average value of a regressor per initialization method.

Section 5.2.5, we examine the effect of raw material, which are encoded as features (see Section 3.1), on the quality.

5.2.1 Choosing a Regressor. We begin by comparing the performance of different regressors. The Multi-layer Perceptron model yielded poor results when compared to the other tested regressors with an average of 401 logAUC over the examined settings. In comparison, linear regression (see Figure 3), has yielded an average logAUC of 290. Thus, we exclude the Multi-layer Perceptron from the following analysis. The inferior results may suggest that when the dataset size is limited, as in the case of active learning, the dominating abilities of deep learning are neutralized.

A comparison between linear regression, Random Forest regression, and XGBoost regression based on logAUC is given in Figure 3. We tested each regressor with a varying number of initialization steps and all initialization methods (Section 4.1), as well as random sampling and a setting with no initial step (total of 252 executions per regressor). Each combination of an initialization method, number of initialization steps, and a selection method is represented as a single dot in the figure.

Figure 3 clearly shows that the random forest and XGBoost regressors outperform linear regression (results are statistically significant). Overall, the performance of random forest and XGboost over CI, Di and Pareto is quite similar (results are not statistically significant). Yet, when looking at the performance over random sampling and no initialization, random forest achieves a statistically significant improvement in performance over XGboost. Additionally, when looking at the best results (lower part of the plot), random forest frequently achieves lower error rates. Lower error rates are obtained by the dominate selection methods (to be evaluated in Section 5.2.3) involving EMCM and QBC (Section 4.2). Henceforth, we shall use random forest regression as the regressor of choice.

5.2.2 Choosing an Initialization Method. We examine the different initialization methods described in Section 4.1, namely, Pareto, Distances (Di), and Clustering (CI). We compare the three methods to random sampling (Random) and no initialization (None). We also analyze the length of the initialization phase in {5,10,15,20,25}. Table 2 compares the MSE of the different initialization methods over time as well as the logAUC (based on which we select the

method, see Section 5.1). The reported results are averaged over the top performing selection methods (to be evaluated in Section 5.2.3).

Table 2 shows all three initialization methods to perform better than random sampling and no initialization. Overall, Di-15 (*i.e.*, applying Distances with 15 steps of initialization) yielded the lowest average logAUC,⁴ and thus, henceforth we use it as our initialization method of choice. Zooming in on the performance of Di and CI until the 15th iteration, CI (and specifically CI-15) achieves a better MSE. The behavior of Di and CI is quite similar until the 40th iteration, after which Di becomes consistently better. It is worth noting that even though Pareto has a slow start, towards the end it achieves comparable MSE to Di.

In Figure 4 we compare the performance of the top performing version of each method with respect to the number of initialization steps over time. The first 10 iterations are noisy regardless of the method. This may be explained by the vulnerability to outliers having a small training set, which characterizes the initial phase. Starting at the 15th iteration, the initialization methods stabilize and consistently decrease. Di-15 seem to converge quickly (around the 60th iteration) and outperform all other methods from that point and on.

5.2.3 Choosing a Selection Method. Next, we analyze the active learning selection methods. Figure 5 illustrates the MSE of the examined selection methods, from which we show the best performing version based on their logAUC (see Figure 6). Specifically, the illustrated methods are random sampling, Expected Model Change Maximization (EMCM_{model}), Predictions (Pr), Uncertainty Mean Squared Error (UMSE), and Query-by-Committee (QBC_{boot}).

As illustrated in Figure 5, QBC and EMCM methods provide a notable improvement in MSE after the initialization phase and outperform the other methods in terms of MSE throughout. This emphasizes the value of committees in selecting useful samples. UMSE performs better than random and consistently decreases over time; yet, unable to compete with QBC and EMCM. This may suggest that uncertainty methods are harder to adopt for regression problems as they typically quantify the certainty of a discrete value. Finally, we note that Pr starts out noisy and stabilizes around the 30th iteration, outperforming both random and UMSE.

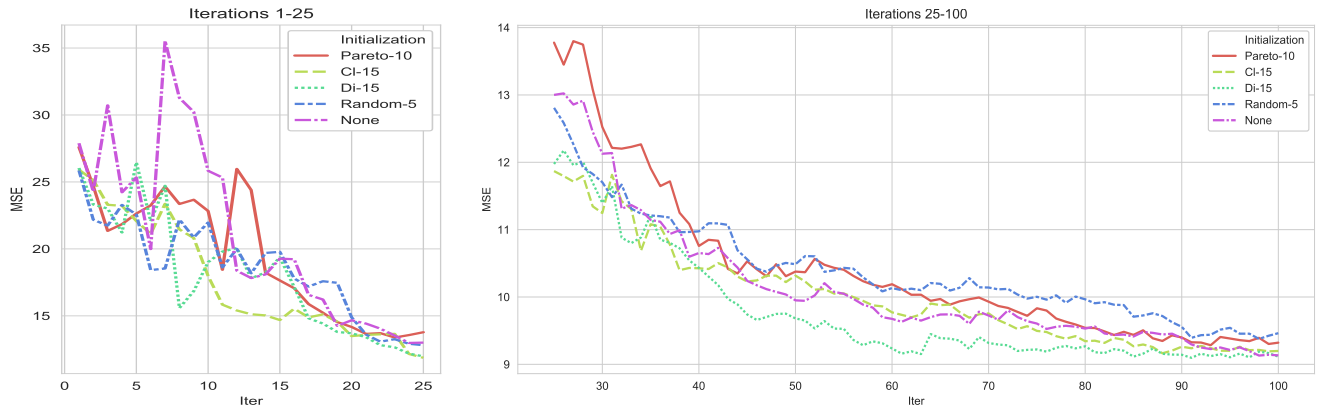
Figure 6 compares the performance of all (initialization and selection) methods in terms of logAUC. Each initialization method (Section 5.2.2) is represented using a separate dot. The non-learning (initialization) methods outperform the (learning) uncertainty-based methods. This serves as another evidence to the difficulty of adapting uncertainty methods to regression methods, suggesting the possible preference of using a heuristic approach, such as Distances or Pareto, for sample selection.

The lowest value in the plot belongs to EMCM_{boot} that uses boot choice of initialization (Section 5.2.2). Yet, overall, we observe that QBC_{boot} performs better on average, and the difference between methods is insignificant. This, in turn, emphasizes the impact of the initialization method on the final outcome. Therefore, a possible direction for a future work involves a careful analysis of the

⁴this result is not statistically significant compared to all methods using Bonferroni correction. Yet, it is statistically significant compared to random and the second best method, CI-15.

Table 2: Comparison of initialization methods (Section 4.1) in terms of MSE over time and logAUC. Statistical significance logAUC performance over random (based on the same initialization steps) is marked with an asterisk

$init(\cdot)$	# steps	5	10	15	20	25	50	75	100	logAUC
Cl	5	22.08	19.41	22.25	12.77	12.04	10.43	9.55	9.30	253.40
	10	22.08	17.98	22.54	13.73	12.23	10.37	9.74	9.18	254.60
	15	22.08	17.98	14.69	13.49	11.87	10.32	9.50	9.20	251.82*
	20	22.08	17.98	14.69	15.54	12.49	10.65	10.00	9.83	256.00
	25	22.08	17.98	14.69	15.54	14.70	10.61	9.88	9.75	256.38
Di	5	26.46	17.76	19.68	14.46	12.96	10.33	9.51	9.25	253.78
	10	26.46	19.01	18.59	14.41	12.90	10.04	9.40	9.00	252.75*
	15	26.46	19.01	19.36	13.70	11.97	9.68	9.22	9.11	250.55*
	20	26.46	19.01	19.36	17.25	11.45	9.59	9.43	9.30	252.54*
	25	26.46	19.01	19.36	17.25	14.59	9.62	9.32	9.47	253.61*
Pareto	5	22.64	20.04	21.89	14.32	13.84	10.35	9.89	9.69	257.39
	10	22.64	22.83	17.64	14.17	13.78	10.38	9.83	9.32	256.39
	15	22.64	22.83	32.12	14.75	13.70	10.57	9.90	9.54	259.74
	20	22.64	22.83	32.12	16.55	12.31	10.10	9.87	9.55	257.75
	25	22.64	22.83	32.12	16.55	15.68	10.45	10.01	9.60	260.33
Random	5	22.56	21.97	19.78	14.89	12.81	10.49	10.00	9.46	255.42
	10	22.56	21.35	25.36	16.33	13.19	10.50	9.63	9.15	257.54
	15	22.56	21.35	20.91	16.42	13.35	10.89	9.69	9.39	259.36
	20	22.56	21.35	20.91	17.69	13.87	10.69	9.68	9.21	258.63
	25	22.56	21.35	20.91	17.69	16.43	10.83	9.81	9.47	260.37
None	0	25.33	25.83	19.27	14.67	13.00	9.95	9.61	9.13	255.23

**Figure 4: Comparison of initialization methods (Section 4.1) over time, where we present the top performing version of each method with respect to the number of initialization steps. The x-axis represents the iteration number and the y-axis the MSE score. The initialization (left) is presented along side an average performance of the different selection methods over the remaining iterations (right).**

combination of initialization and selection methods when applying active learning techniques.

5.2.4 Analyzing the Quality of Learning. Sections 5.2.1-5.2.3 focused on MSE and logAUC as common practice measures of success for active learning [5]. Next, we examine other aspects of active learning, following the discussion in Section 4.3.

Table 3 compares all (initialization and selection) methods, using the performance measures of mean squared error (MSE) (with 20,

50, and 100 iterations), area under curve (AUC) and its logarithmic counterpart (logAUC), smoothness as determined by absolute second difference (ASD) and weighted absolute second difference (WASD), and convergence (FTC).

In Section 5.2.3, we showed QBC and ECM methods to have superior performance, with final MSE values ranging between 8.41 (QBC_{boot}) and 9.88 (ECM_{model}) and logAUC between 245.87 (QBC_{boot}) and 251.7 (ECM_{model}). In addition, all QBC and ECM methods perform significantly better than random. Yet, when it comes to smoothness and convergence, random selection

Table 3: Comparison among all methods (including both initialization and selection) in terms of MSE, AUC, logAUC, ASD, WASD, and FTC (Section 4.3). Statistical significance of logAUC performance over random (based on the same initialization steps) is marked with an asterisk.

Method		MSE (20)	MSE (50)	MSE (100)	AUC	logAUC	ASD	WASD	FTC
Initialization Methods	CI	16.39	10.72	10.25	1315.11	261.76	0.98	0.34	97.00
	Di	17.25	10.10	9.49	1267.56	257.34*	1.14	0.42	99.00
	Pareto	18.33	10.70	10.22	1294.59	259.56	1.12	0.46	96.00
	Pr	17.77	10.52	9.53	1278.00	257.50*	1.40	0.54	93.00
Uncertainty Methods	UCL	17.54	11.25	10.56	1336.36	263.33	0.92	0.31	99.00
	UDi	20.54	12.14	10.96	1430.77	270.07	1.15	0.46	100.00
	UMSE	15.56	10.89	9.90	1295.26	260.08	0.90	0.31	99.00
Expected Model Change Maximization	EMCM _{boot}	13.01	8.72	8.67	1127.48	245.87*	0.98	0.38	100.00
	EMCM _{model}	12.63	10.65	9.88	1186.87	251.70*	0.91	0.34	100.00
Query by Committee	QBC _{boot}	13.52	9.64	8.41	1164.28	249.33*	0.93	0.35	99.00
	QBC _{model}	12.08	9.30	9.13	1154.09	248.53*	0.89	0.29	94.00
	Random	17.95	11.11	10.76	1342.70	263.78	0.72	0.17	64.00

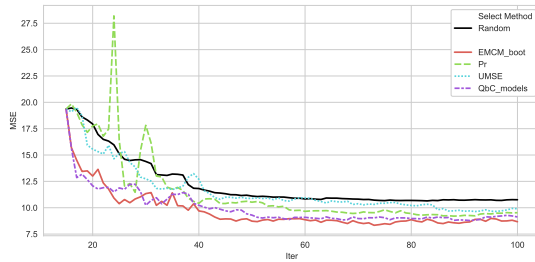


Figure 5: Comparison between selection methods (Section 4.2) over time using a random forest regressor and 15 initialization steps using Di. For each method we show the version providing the lowest logAUC (see Figure 6). The x-axis represents the iteration number (starting after the initialization) and the y-axis the MSE score.

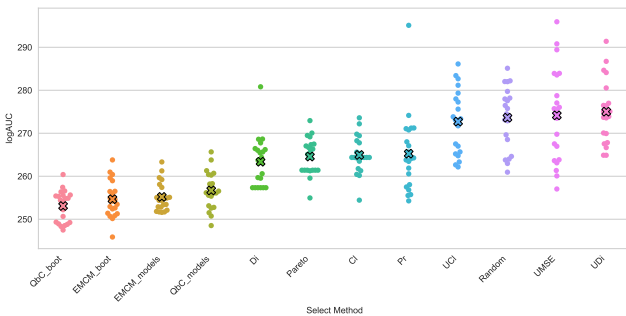


Figure 6: Comparison between selection methods (Section 4.2) using logAUC. Table 3 provides statistical significance results

provides a smooth learning curve that rapidly converges (64th iteration compared to other methods that converge just before the

finish line). This dominance, in terms of smoothness and convergence, can be explained by the inability of random to improve on its performance, smoothly converging to a poor result fast. Additionally, random was also averaged over 15 runs, which may also affect its smoothness. Interestingly, the non-random methods, UCL and UMSE are only second to QBC_{model} in terms of WASD. This indicates that although using uncertainty methods for regression results in inferior performance (compared to QBC and EMCM) it does provide a smooth learning curve. Finally, all non-random methods coverage roughly at the same point towards the end of the experiment. This serves as evidence that additional training data (either in a supervised learning setup or by extending the experiment horizon) can further improve on the reported results.

5.2.5 Feature Analysis. In our final analysis we recall one of the core goals of this work, namely understanding the impact of cow and milk features on cottage cheese production. To do so, we provide feature importance analysis using SHAP [41], a state-of-the-art technique to interpret machine learning models. The top 15 important features are given in Figure 7 ranked by their shap values, representing their relative importance in the trained model.

The most dominant feature based on our analysis is *Oa*. *Oa* together with *Pufa* (3rd) and *Mufa* (5th) represent the dynamics of milk coagulation process (see Example 3.1) and higher values affect CF positively. This, in turn, provides a new insight to the milk industry as no prior research has considered these properties as predictors of quality. The enzymatic coagulation of milk (*Casein* 2-nd), conductivity, and lactose (4-th) validate our analysis as they are known to have an effect on quality [30, 36, 44]. Here, *Casein* is more significant and higher values have a positive impact on CF. Note that, although lactose mostly has a positive effect, there are a few outliers with opposite effect.

6 RELATED WORK

There are numerous machine (and deep) learning solutions for the food industry domain [28, 37, 62] and other domains with similar characteristics, such as transportation [18, 51] and air pollution [23,

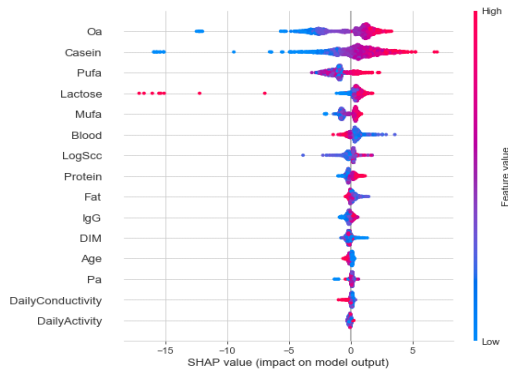


Figure 7: Feature importance using SHAP

52]. Common to all of these works is the availability of labeled data. We focus on a setting where labeled data is scarce, applying active learning to efficiently utilize available data samples.

Active learning methods prioritize data to be labeled in order to maximize the models ability to learn. Active learning is widely used for classification problems [11, 31, 33, 56] and was also used for several data management tasks. Arasu *et al.* [3] consider the task of record matching packages in an active learning setting and Mozafari *et al.* [48] use active learning for an optimization strategy in crowd-sourced databases. More recently, the tasks of interactive database exploration [24], time series and text classification [13, 38], enhancing database systems [42] have also been addressed with active learning techniques. In this work we use active learning for regression, which, has yet to gain wider popularity.

Aiming to use active learning for regression, Burbidge *et al.* suggests using a committee of classifiers (or regressors) on different subsets of the labeled data [10]. Cai *et al.* presents a data sampling solution in the context of regression, which queries the examples that lead to the largest model change [11]. Finally, O'Neill compares active learning techniques in the context of linear regression models [50]. In this work, we compare the aforementioned methods and others in our unique scenario of the milk industry.

Consensus is not yet apparent regarding the evaluation of active learning methods. Several works base their analysis on qualitative criteria [25, 57, 60]. Others focus on quantitative evaluation [56]. Specifically, Baram *et al.* suggests to evaluate the performance of an active learner over time using the AUC measure [5]. In our analysis, we use the logarithmic AUC as our leading comparison measure among methods. We also adapt additional evaluation measures, for example ASD and FTC, to account for additional perspectives of active learning.

Finally, several previous works analyze cheese-milk quality [30, 36, 44]. Leitner *et al.* investigates the effect of milk storage on quality using individual cows. Katz *et al.* examines the importance of milk composition for the final product [30]. In our empirical evaluation we validate some of the findings of these studies, enriching the literature with new insights that demonstrate the dominance of dynamics of milk coagulation process as quality predictors.

7 CONCLUSIONS

In this work we focus on the problem of acquiring a training dataset with limited resources. We offer a data collection schedule methodology, based on active learning, to predict production quality using a limited amount of annotated raw material data. We demonstrate the usefulness of the approach with a real-world use-case in the milk industry, where we use milk and cow features to predict the quality of cottage cheese.

Our suggested algorithmic solution is composed of two main phases handling varying sizes of available training data. First, with a small training set, we utilize human meta-knowledge to identify useful samples. Then, as the size of labeled training data set increases, we extend state-of-the-art methods of active learning to handle regression-based problems.

An extensive empirical evaluation that uses data collected at Afimilk laboratory demonstrates the effectiveness of our proposed approach. The results indicate that when the training set is limited, random forest regressor preforms better than linear, Multi-layer Perceptron neural network, and XGBoost regressors, the active learning benefits from an initialization phase, and committee-based selection (expected model change maximization and query by committee) outperforms other selection methods. Feature analysis reveals new insights, indicating that the dynamics of milk coagulation process impact the production quality.

The suggested active learning framework was developed for (and based on) a real-world setting of a cheese production line. The presented framework is applicable to any IoT scenario where: 1) labeled data is scarce and hard or expensive to obtain, 2) prediction is needed for a process outcome (using regression rather than classification), and 3) the pool of samples continuously changes. As part of our ongoing research we aim to experiment with additional industrial domains such as viticulture.

In future work we also intend to analyze the active learning methodologies in additional settings. This includes combining various active learning methods in a single experiment, analyze the impact of the budget and experiment horizon, taking advantage of the relationship between available sample sets (*e.g.*, same cow at different timestamps), and designing a domain specific selection methodology directly aiming at maximizing one or more of the suggested quality measures. In addition, in our experiments we used individual cow samples. We aim to extend our framework to handle varying granularity levels, including milk coming from different cows in the same shed (containing several indistinguishable cows) and cows from multiple farms.

REFERENCES

- [1] [n.d.]. Repository. <https://github.com/nitaytech/ActiveLearningForRegression>.
- [2] [n.d.]. Technical Report. <https://github.com/nitaytech/ActiveLearningForRegression>.
- [3] Arvind Arasu, Michaela Götz, and Raghav Kaushik. 2010. On active learning of record matching packages. In *SIGMOD*. 783–794.
- [4] H. V. Atherton and J. A. Newlander. 1977. *Chemistry and testing of dairy products*. AVI Publishing Co. Inc.
- [5] Yoram Baram, Ran El Yaniv, and Kobi Luz. 2004. Online choice of active learning algorithms. *Journal of Machine Learning Research* 5, Mar (2004), 255–291.
- [6] HAR Barnett. 1985. Criteria of smoothness. *Journal of the Institute of Actuaries* 112, 3 (1985), 331–367.
- [7] D. G. Blackburn, V. Hayssen, and C.J. Murphy. 1989. The origins of lactation and the evolution of milk: a review with new hypotheses. *Mammal Review* 19, 1 (1989), 1–26.

- [8] Léon Bottou. 2010. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*. Springer, 177–186.
- [9] R. J. Brown, C. A. Ernstrom, and M. E. Johnson. 1988. *Milk-Clotting Enzymes and Cheese Chemistry*. Springer, 609–654.
- [10] Robert Burbidge, Jem J Rowland, and Ross D King. 2007. Active learning for regression based on query by committee. In *International conference on intelligent data engineering and automated learning*. Springer, 209–218.
- [11] Wenbin Cai, Ya Zhang, and Jun Zhou. 2013. Maximizing expected model change for active learning in regression. In *ICDM*. IEEE, 51–60.
- [12] Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *SIGKDD*. 785–794.
- [13] Gordon V Cormack and Maura R Grossman. 2016. Scalability of continuous active learning for reliable high-recall text classification. In *Proceedings of the 25th ACM international conference on information and knowledge management*. 1039–1048.
- [14] Ido Dagan and Sean P Engelson. 1995. Committee-based sampling for training probabilistic classifiers. In *Machine Learning Proceedings 1995*. Elsevier, 150–157.
- [15] David I Doupchrade, G Robert Hagevoort, Matthew W Nonnenmann, Christina Lunner Kolstrup, Stephen J Reynolds, Martina Jakob, and Mark Kinsel. 2013. The dairy industry: a brief description of production practices, trends, and farm characteristics around the world. *Journal of agromedicine* 18, 3 (2013), 187–197.
- [16] TF Dunlap, RA Kohn, GE Dahl, M Varner, and RA Erdman. 2000. The impact of somatotropin, milking frequency, and photoperiod on dairy farm nutrient flows. *Journal of Dairy Science* 83, 5 (2000), 968–976.
- [17] NC Friggens, GC Emmans, I Kyriazakis, JD Oldham, and M Lewis. 1998. Feed intake relative to stage of lactation for dairy cows consuming total mixed diets with a high or low ratio of concentrate to forage. *Journal of dairy science* 81, 8 (1998), 2228–2239.
- [18] Avigdor Gal, Avishai Mandelbaum, François Schnitzler, Arik Senderovich, and Matthias Weidlich. 2017. Traveling time prediction in scheduled transportation with journey segments. *Information Systems* 64 (2017), 266–280.
- [19] W.M. Gelbart, A Ben-Shaul, and D ROUX (Eds.). 1994. *Micelles, Membranes, Microemulsions and Monolayers*. Springer, New York.
- [20] W. M. Gelbart and A. Ben-Shaul. 1996. The “New” Science of “Complex Fluids”. *The Journal of Physical Chemistry* 100, 31 (1996), 13169–13189.
- [21] D. Gianola and G. J. M. Rosa. 2015. One Hundred Years of Statistical Developments in Animal Breeding. *Annual Review of Animal Biosciences* 3, 1 (2015), 19–56.
- [22] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. 249–256.
- [23] Philipp Haehnel, Jakub Mareček, Julien Monteil, and Fearghal O'Donncha. 2020. Using deep learning to extend the range of air pollution monitoring and forecasting. *J. Comput. Phys.* 408 (2020), 109278.
- [24] Enhui Huang, Liping Peng, Luciano Di Palma, Ahmed Abdelkafi, Anna Liu, and Yanlei Diao. [n.d.]. Optimization for Active Learning-based Interactive Database Exploration. *Proceedings of the VLDB Endowment* 12, 1 ([n. d.]).
- [25] Sheng-Jun Huang, Rong Jin, and Zhi-Hua Zhou. 2010. Active learning by querying informative and representative examples. In *Advances in neural information processing systems*. 892–900.
- [26] K. L. Ingvarsten, O. Aes, and J. B. Andersen. 2001. Effects of pattern of concentrate allocation in the dry period and early lactation on feed intake and lactational performance in dairy cows. *Livestock Production Science* 71, 2-3 (2001), 207–221.
- [27] R. Jensen. 2002. Invited Review: The Composition of Bovine Milk Lipids: January 1995 to December 2000. *Journal of Dairy Science* 85, 2 (2002), 295–350.
- [28] Ana M Jiménez-Carvelo, Antonio González-Casado, M Gracia Bagur-González, and Luis Cuadros-Rodríguez. 2019. Alternative data mining/machine learning methods for the analytical evaluation of food quality and authenticity—A review. *Food research international* 122 (2019), 25–39.
- [29] C.T. Kadzere, M.R. Murphy, N. Silanikov, and E. Maltz. 2013. Heat stress in lactating dairy cows: a review. *Livestock Production Science* 77, 1 (2013), 59–91.
- [30] G. Katz, U. Merin, D. Bezman, S. Lavie, L. Lemberskiy-Kuzin, and G. Leitner. 2016. Real-time evaluation of individual cow milk for higher cheese-milk quality with increased cheese yield. *Journal of Dairy Science* 99, 6 (2016), 10587–10598.
- [31] Mahnoosh Kholghi, Laurianne Sitbon, Guido Zuccon, and Anthony Nguyen. 2015. External knowledge and query strategies in active learning: a study in clinical information extraction. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management*. 143–152.
- [32] I. Kubarsepp, M. Henno, O. Kart, and T. Tupasela. 2005. A comparison of the methods for determination of the rennet coagulation properties of milk. *Acta Agriculturae Scandinavica Section A – Animal Science*, 55 (2005), 145–148.
- [33] Ken Lang. 1995. Newsweeder: Learning to filter netnews. In *Machine Learning Proceedings 1995*. Elsevier, 331–339.
- [34] Juha Lappi. 2006. Smooth height/age curves from stem analysis with linear programming. (2006).
- [35] B.A. Law and A.Y. Tamime. 2010. *Technology of Cheesemaking*. Blackwell Publishing Ltd.
- [36] Gabriel Leitner, Nissim Silanikove, Shamay Jacobi, Limor Weisblit, Solange Bernstein, and Uzi Merin. 2008. The influence of storage on the farm and in dairy silos on milk quality for cheese production. *International Dairy Journal* 18, 2 (2008), 109–113.
- [37] Bing Li, Yinzi Lin, Wei Yu, David I Wilson, and Brent R Young. 2020. Application of mechanistic modelling and machine learning for cream cheese fermentation pH prediction. *Journal of Chemical Technology & Biotechnology* (2020).
- [38] Shen Liang, Yanchun Zhang, and Jiangang Ma. 2020. Active model selection for positive unlabeled time series classification. In *ICDE*. IEEE, 361–372.
- [39] Andy Liaw, Matthew Wiener, et al. 2002. Classification and regression by random Forest. *R news* 2, 3 (2002), 18–22.
- [40] J.A. Lucey, M.E. Johnson, and D.S. Horne. 2003. Invited Review: Perspectives on the Basis of the Rheology and Texture Properties of Cheese. *Journal of Dairy Science* 86, 9 (2003), 2725–2743.
- [41] Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in neural information processing systems*. 4765–4774.
- [42] Lin Ma, Bailu Ding, Sudipto Das, and Adith Swaminathan. 2020. Active learning for ML enhanced database systems. In *SIGMOD*. 175–191.
- [43] Yinqing Ma, C J Ryan, DM Barbano, DM Galton, MA Rudan, and KJ Boor. 2000. Effects of Somatic Cell Count on Quality and Shelf-Life of Pasteurized Fluid Milk1. *Journal of dairy science* 83, 2 (2000), 264–274.
- [44] I. Q. Macedo, C. J. Faro, and E. M. Pires. 1993. Specificity and kinetics of the milk-clotting enzyme from cardoon (*Cynara cardunculus* L.) toward bovine kappa-casein. *Journal of Agricultural and Food Chemistry* 41, 10 (1993), 1537–1540.
- [45] James MacQueen et al. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, Vol. 1. Oakland, CA, USA, 281–297.
- [46] E. Maltz. 2010. Novel Technologies: Sensors, Data and Precision Dairy Farming. *The first North American conference on precision dairy management* (2010).
- [47] D. J. McMahon and R. J. Brown. 1982. Evaluation of Formagraph for Comparing Rennet Solutions. *Journal of dairy science* 65, 8 (1982), 1639–1642.
- [48] Barzan Mozafari, Purna Sarkar, Michael Franklin, Michael Jordan, and Samuel Madden. 2014. Scaling up crowd-sourcing to very large datasets: a case for active learning. *Proceedings of the VLDB Endowment* 8, 2 (2014), 125–136.
- [49] G. L. Munro, P. A. Grieve, and B. J. Kitchen. 1984. Effects of mastitis on milk yield, milk composition, processing properties and yield and quality of milk products. *Australian Journal of Dairy Technology* 39, 1 (1984), 7–16.
- [50] Jack O'Neill. 2015. An evaluation of selection strategies for active learning with regression. (2015).
- [51] Niklas Christoffer Petersen, Filipe Rodrigues, and Francisco Camara Pereira. 2019. Multi-output bus travel time prediction with convolutional LSTM neural network. *Expert Systems with Applications* 120 (2019), 426–435.
- [52] Zhongang Qi, Tianchun Wang, Guojie Song, Weisong Hu, Xi Li, and Zhongfei Zhang. 2018. Deep air learning: Interpolation, prediction, and feature analysis of fine-grained air quality. *TKDE* 30, 12 (2018), 2285–2297.
- [53] M. Ron and J. Weller. 2007. From QTL to QTN identification in livestock – winning by points rather than knock-out: a review. *Animal Genetics* 38, 5 (2007), 429–439.
- [54] Peter J Rousseeuw. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics* 20 (1987), 53–65.
- [55] George AF Seber and Alan J Lee. 2012. *Linear regression analysis*. Vol. 329. John Wiley & Sons.
- [56] Burr Settles. 1995. Active Learning Literature Survey. *Science* 10, 3 (1995), 237–304.
- [57] Dan Shen, Jie Zhang, Jian Su, Guodong Zhou, and Chew-Lim Tan. 2004. Multi-criteria-based active learning for named entity recognition. In *ACL*. Association for Computational Linguistics, 589.
- [58] G. E. Shook. 2006. Major Advances in Determining Appropriate Selection Goals. *Journal of Dairy Science* 89, 4 (2006), 1349–1361.
- [59] J. I. Weller and E. Ezra. 2016. Genetic and phenotypic analysis of daily Israeli Holstein milk, fat, and protein production as determined by a real-time milk analyzer. *Journal of Dairy Science* 99, 12 (2016), 9782–9795.
- [60] Dongrui Wu. 2018. Pool-based sequential active learning for regression. *IEEE transactions on neural networks and learning systems* 30, 5 (2018), 1348–1359.
- [61] Yanjun Yao, Qing Cao, and Athanasios V Vasilakos. 2014. EDAL: An energy-efficient, delay-aware, and lifetime-balancing data collection protocol heterogeneous wireless sensor networks. *IEEE/ACM transactions on networking* 23, 3 (2014), 810–823.
- [62] DaeYoung Yoon and Simon S Woo. 2020. Who is Delivering My Food? Detecting Food Delivery Abusers using Variational Reward Inference Networks. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2917–2924.