University of Zurich

Implementing Correlation Measures for Streaming Time Series

Master-Project

Marc-Alain Chételat 10-915-718
Moritz Schneider 11-490-455

Database Technology
Prof. Dr. Michael Böhlen

Project
supervised by
Kevin Wellenzohn

March 6, 2017

Abstract

We study the problem of ranking a set of (reference) time series according to their similarity to a (base) time series. Creating such a ranking is an important subroutine for many algorithms, e.g. TKCM exploits ranked reference time series to impute missing values in a (base) time series. In this report, we describe two similarity measures, called the Pearson correlation coefficient (PCC) and Case Matching Similarity (CMS), for computing the similarity between time series. Both measures have been implemented and tested on a large real-world data set of meteorological time series that we collected from MeteoSwiss. An experimental evaluation has shown that the similarity of two time series depends heavily on their length. In particular, the ranking of the reference time series fluctuates strongly for short time series. We found three interesting weather phenomena in the MeteoSwiss data set: the Föhn, the Bise, and the temperature inversion. The experiments with the weather phenomena have shown that rather short periods of time in which the weather phenomena occur are too short to have a high impact on the PCC or CMS, because a strong linear correlation exists in the remaining period.

Contents

List of Figures

List of Tables

List of Algorithms

1 Introduction

A streaming time series s is a sequence of data points that receives a new value every time unit (e.g. every 10 minutes). Such data appears in many applications, e.g. the financial stock market, meteorology, sensor networks, and network monitoring to name only a few. Often streams are incomplete as values are missing.

Top-k Case Matching (TKCM) is an algorithm to impute missing values in streams of meteorological time series, i.e. to replace the missing value with good estimates of what the values could have been [1]. TKCM imputes a missing value at time t of (base) time series s by looking for similar values to those seen at a set of reference time series at time t. TKCM assumes that a ranking of reference time series for a time series s is provided by domain experts.

This paper focuses on two similarity (or correlation) measures to automatically create a ranking of reference time series. The first measure is the Pearson Correlation Coefficient (PCC), which measures the linear correlation between two time series. Informally, two time series are linearly correlated when their scatter plot shows a linear trend. We prove that PCC reaches its maximum value if the scatter plot shows a perfectly straight line.

The second measure is the Case Matching Similarity (CMS), which is also able to detect non-linear correlation between time series. Time series shifted in the time axis are an example of non-linearly correlated time series. As for PCC, we also show for CMS under which circumstances two time series are maximally correlated.

The paper is structured as follows: In Section 2, we introduce the basic notation and define the goal of our work. Section 3 introduces the MeteoSwiss data set which has been used during an experimental evaluation. Section 4 introduces the PCC and shows formally the connection between PCC and linear regression. More specifically, it shows that PCC can be interpreted as quality measure for the obtained regression line. Section 5 introduces the CMS and discusses its properties; for instance, we show that CMS is asymmetric, i.e. $CMS_w(r, s) \neq CMS_w(s, r)$ for two time series s, r. This stems from the fact that in TKCM a time series r can be a good reference for time series s, but not vice versa. In addition, a formula was specified to provide an interval for the CMS's bucket width. Section 6 presents pseudocode of the implementation to compute PCC and CMS incrementally on streams of time series. Furthermore, a complexity analysis of the implementation is given.

We continue with the experimental evaluation in Section 7 which comprises an experimental evaluation including the impact of the sliding window on the ranking and the impact of the bucket width on the ranking. In addition, Section 7.2 explains three weather phenomena that can be found in our data set and we test our implementation during their occurrence. Section 8 concludes our work and presents future research directions.

2 Problem Statement

We consider a set $\mathbf{S} = \{s_1, s_2, \dots, s_n\}$ of n streaming time series. Each time series $s \in \mathbf{S}$ reports values from a sensor measured at time points $t_1, t_2, \dots,$ where $s(t)$ denotes the value of s at time t. Since streaming time series are unbounded, we introduce a sliding window W of length L_W. The sliding window $W = \{t_{n-L_W+1}, \dots, t_{n-1}, t_n\}$ contains the L_W last time points of the measurements that are kept in main memory where t_n is the current time. The aim of our work is to determine for some (base) time series $s \in \mathbf{S}$ a ranking of the remaining time series $\mathbf{R} = \mathbf{S} \backslash \{s\}$. We call a time series $r \in \mathbf{R}$ a reference time series. To establish a ranking, we use a similarity function $f(s, r)$ that computes the similarity (or correlation) between a base time series s and a reference time series $r \in \mathbf{R}$. Let $r_1, r_2 \in \mathbf{R}$ be two reference time series, we say that r_1 is more similar to s than r_2 to s if and only if $f(s, r_1) \succcurlyeq f(s, r_2)$.

Definition 1. (*Ranking*) Let s be a time series and let f be a similarity function. A ranking of reference time series $\langle r_{i_1}, r_{i_2}, \dots, r_{i_{n-1}} \rangle$ is an ordered sequence of time series $r_i \in \mathbf{R}$ such that $f(s, r_{i_j}) \succcurlyeq f(s, r_{i_{j+1}})$ for any $j \in [1, n-1)$.

The goal of our work is to study two similarity function f: the Pearson Correlation Coefficient (PCC) [2] and the Case Matching Similarity (CMS) [3]. PCC is able to detect linear correlation, while CMS is able to detect also non-linear correlation. The PCC ranges from -1 to 1, where a higher absolute value denotes stronger correlation, hence $PCC(s, r_1) \succcurlyeq PCC(s, r_2) \iff |PCC(s, r_1)| \geq |PCC(s, r_2)|$. The CMS range is $[0, \infty)$, with 0 denoting maximum similarity, hence $CMS_w(s, r_1) \succcurlyeq CMS_w(s, r_2) \iff CMS_w(s, r_1) \leq CMS_w(s, r_2)$. Table 1 summarizes the notations used in this paper.

$\mathbf{S}$	Set of all time series
$\mathbf{R}$	Set of all reference time series
s	Base time series
r	Reference time series
t	Time of measurement
$s(t)$	The measured value at time t in time series s
W	Sliding Window
L_W	The maximum length of the sliding window W
b_z	A bucket, where z denotes the ID of the bucket and $z \in \mathbb{Z}$
w	Bucket width, where $w \in \mathbb{R}_{>0}$
σ_z	Bucket standard deviation
$\bar{b}_z$	Bucket mean
c	Average rate of change

Table 1: Summary of main notations.

3 The MeteoSwiss Data Set

The *Federal Office of Meteorology and Climatology MeteoSwiss* operates 1268 weather stations in Switzerland, each station records several meteorological parameters in variable intervals. Temperature data, sunshine duration, rainfall, air pressure data, and many other parameters are recorded in a one, five, ten, 15 minutes and hourly interval. Our meteorological data set comprises 117 time series, each ten years long – from 2006 to 2015. The data set sources from 80 different weather stations mainly located in the cantons Wallis, Graubünden and the Rhone-valley. Since weather phenomena like the Föhn occurs in valleys close to the Alps, these region's weather data suits best for our analysis. The data set contains 40,698,841 measurements in total, where the shortest and longest time series contain 13,353 and 525,888 measurements respectively. 20,830,055 values in the data set are missing (33.85%). Missing values often occur in groups, i.e. blocks of consecutive missing values. Small blocks are mainly caused by transmission problems while larger occur due to sensor failures. Around 62% of the data set are temperature measurements. The temperature is measured two meters above ground. 28% of the data set consist of barometric pressure and the rest contains measurements about absolute humidity. The lowest temperature measured was -35.1 °C in Samedan in February 2012, 39.7 °C in Genève-Cointrin was the highest measured temperature respectively. The lowest barometric pressure was measured on the Piz Corvatsch with 641.5hPa. The highest measured barometric pressure in the timeframe from 2006 to 2015 was measured in Genève-Cointrin with 990.5hPa. The lowest absolute humidity was 0 g/m^3 - the highest absolute humidity was measured in Sitten with 18.4 g/m^3.

Figure 1 and 2 show the distribution of all chosen weather stations. The northernmost station is Güttingen. The easternmost station is Scuol in the area called Engadin. The southernmost station is called Grosser Sankt Bernhard in the canton Wallis and the westernmost Genève-Cointrin. The weather stations have common parameters such as they are located in the same valley or on the same altitude above sea level. The station Grono in the canton Graubünden has with 323m above sea level the lowest altitude. Whereas the station on the Piz Corvatsch, also located in canton Graubünden, is 3302m above sea level.

3.1 Sample Data Set

The sample data set is a subset of our meteorological data set. It will be used in this paper for applying formulas and algorithms. Table 2 defines a

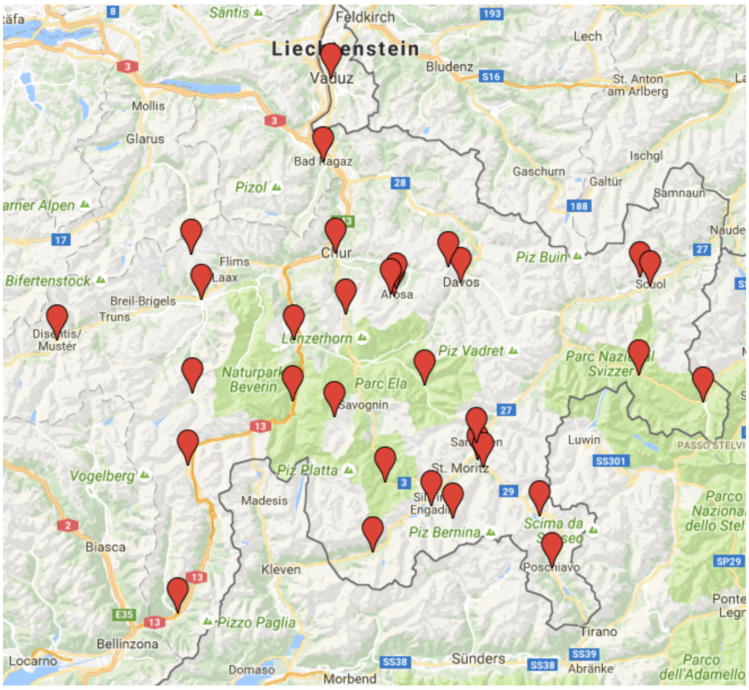

Figure 1: Weather stations in the canton Graubünden and Vaduz.

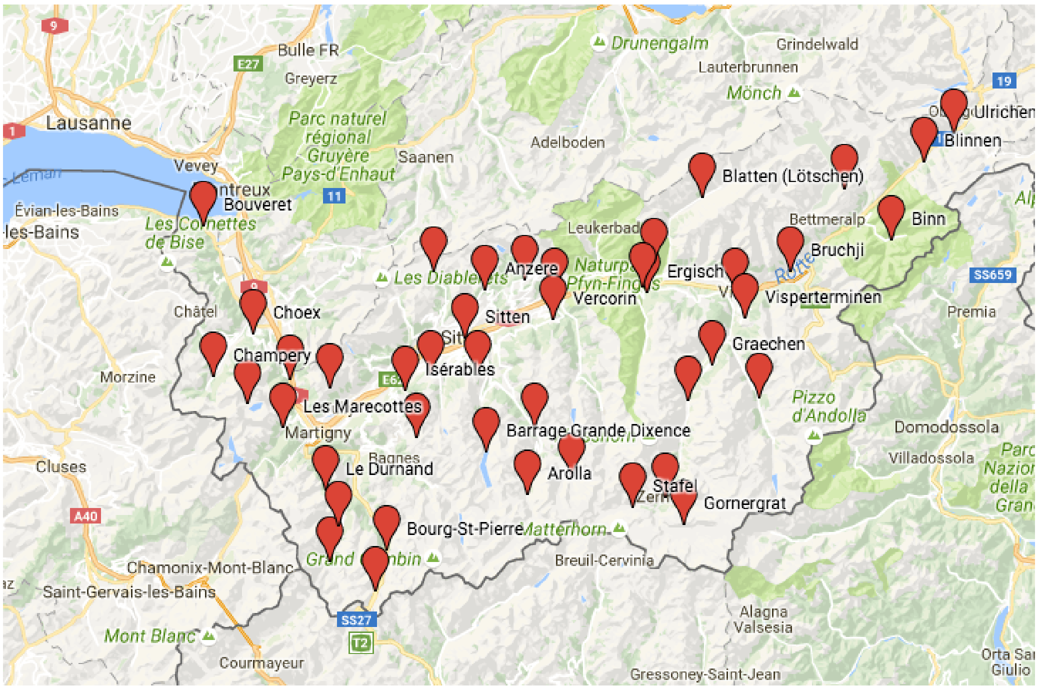

Figure 2: Weather stations in the canton Wallis and the Rhone-valley.

set $S = \{r_1, s_1\}$ of time series.

Time $t \in W$	$r_1(t)$ $°C$	$s_1(t)$ $°C$
1 (22:00)	1.3	7.9
2 (22:20)	0.6	7.5
3 (22:40)	0.5	7.7
4 (23:00)	0.1	7.4
5 (23:20)	0	8
6 (23:40)	0.2	5.9
7 (00:00)	0	6.3
8 (00:20)	-0.5	5.8
9 (00:40)	-0.8	5.3
10 (01:00)	-1.2	5.1
11 (01:20)	-1.4	5.5
12 (01:40)	-1.2	5
13 (02:00)	-1.1	5.6
Mean	-0.269	6.385

Table 2: Weather data from Siders (r_1) and Visp (s_1) while the Föhn occurs (14/15.03.15).

4 Pearson Correlation Coefficient

The Pearson Correlation Coefficient (PCC), also product moment correlation coefficient, is a measure of the linear correlation between two variables, giving a value between +1 and -1 inclusive, where 1 is a total positive and -1 a total negative linear correlation respectively. In the case the scatter plot of the two variables (time series in our case) shows a perfectly rising or falling line, as we will prove in this section. PCC approaches zero the more values differ from the straight line. If the two variables do not linearly correlate at all, the PCC's value is zero. First we define PCC and calculate then various quantities using the data set defined in Section 3. Later on we formally show the connection between PCC and linear regression. More specifically, we show that PCC can be interpreted as a quality measure for the obtained regression line.
Let r and s be two time series then PCC is defined as

$$PCC(s,r) = \frac{\sum_{t \in W}(r(t) - \bar{r})(s(t) - \bar{s})}{\sqrt{\sum_{t \in W}(r(t) - \bar{r})^2}\sqrt{\sum_{t \in W}(s(t) - \bar{s})^2}} \tag{1}$$

where

$$\bar{s} = \frac{\sum_{t \in W} s(t)}{L_w} \tag{2}$$

is the mean of s, r respectively [2]. We wish to measure both the *direction* and the *strength* of the linear relationship between s and r. First we develop the covariance and then the correlation coefficient (Equation 1). Let us draw a horizontal line at $\bar{s}_1$ and a vertical line at $\bar{r}_1$ on the scatter plot (Figure 3) of the two time series s_1 and r_1 of our running example. For our data set the means are

$$\bar{s}_1 = \frac{\sum_{t \in W} s_1(t)}{L_w} = \frac{83}{13} = 6.385 \text{ and } \bar{r}_1 = \frac{\sum_{t \in W} r_1(t)}{L_w} = \frac{-3.5}{13} = -0.269.$$

The two lines divide the scatter plot into four quadrants. The following quantities are computed in Table 3 for each time point $t \in W$:

- $(s_1(t) - \bar{s}_1)$, the deviation of each observation $s_1(t)$ from $\bar{s}_1$,
- $(r_1(t) - \bar{r}_1)$, the deviation of each observation $r_1(t)$ from $\bar{r}_1$,
- the product of the above two quantities, $(s_1(t) - \bar{s}_1)(r_1(t) - \bar{r}_1)$.

The quantity $(s_1(t) - \bar{s}_1)$ is positive for every point in the first and second quadrants, and it is negative for every point in the third and fourth quadrants. Similarly, the quantity $(r_1(t) - \bar{r}_1)$ is positive for every point in the first and

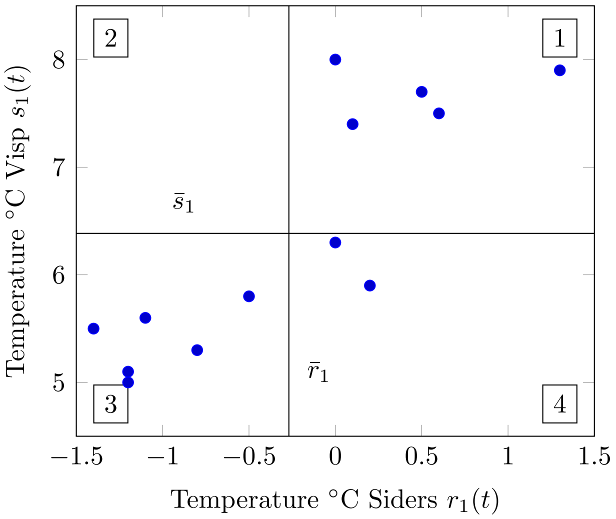

Figure 3: Data set's (Table 2) scatter plot indicating $\bar{s}_1$, $\bar{r}_1$ and the quadrants.

$t \in W$	$(s_1(t) - \bar{s}_1)$	$(r_1(t) - \bar{r}_1)$	$(s_1(t) - \bar{s}_1)(r_1(t) - \bar{r}_1)$
1	1.515	1.569	2.378
2	1.115	0.869	0.970
3	1.315	0.769	1.012
4	1.015	0.369	0.375
5	1.615	0.269	0.435
6	-0.485	0.469	-0.227
7	-0.085	0.269	-0.023
8	-0.585	-0.231	0.135
9	-1.085	-0.531	0.576
10	-1.285	-0.931	1.196
11	-0.885	-1.131	1.000
12	-1.385	-0.931	1.289
13	-0.785	-0.831	0.652
Sum	0.000	0.000	9.766

Table 3: Quantities needed to compute covariance between the temperature series of Visp (s_1) and Siders (r_1).

fourth quadrants, and it is negative for every point in the second and third quadrants. As Table 3 shows the sums of $(s_1(t) - \bar{s}_1)$ and $(r_1(t) - \bar{r}_1)$ equal zero. Since the quantities reflect the distance to the mean the sums are supposed to be zero. If the linear relationship between s_1 and r_1 is positive (as s_1 increases also r_1 increases), then there are likely more points in the first and third quadrants than in the second and fourth quadrants. In this case, the sum of the product $(s_1(t) - \bar{s}_1)(r_1(t) - \bar{r}_1)$ is likely to be positive because there are more positive than negative quantities. Conversely, if the relationship between s_1 and r_1 is negative (as s_1 increases, r_1 decreases), then there are likely more points in the second and fourth quadrants than in the first and third quadrants. Hence the product $(s_1(t) - \bar{s}_1)(r_1(t) - \bar{r}_1)$ is likely to be negative. If we compare with our data set, all data points lie in the first and third quadrants except the time points $t = 6, 7$ lying in the fourth quadrant which results in a negative sign of the product $(s_1(t) - \bar{s}_1)(r_1(t) - \bar{r}_1)$ for $t = 6, 7$ (see Table 3). Therefore the sign of the quantity

$$Cov(s, r) = \frac{\sum_{t \in W}(s(t) - \bar{s})(r(t) - \bar{r})}{L_w} \tag{3}$$

which is known as the *covariance* between r and s, indicates the direction of the linear relationship between s and r. If $Cov(s, r) > 0$, then there is a positive relationship between s and r, if $Cov(s, r) < 0$, then the relationship is negative. If $Cov(s, r) = 0$, then there is no linear relationship between s and r [2].

Since the sum of the products in Table 3 is positive for our data set we expect $Cov(s_1, r_1) > 0$ and therefore a positive linear relationship between the time series s_1 and r_1 (i.e. as the temperature in Siders increases also the temperature in Visp increases) which is the case applying Equation 3:

$$Cov(s_1, r_1) = \frac{\sum_{t \in W}(s_1(t) - \bar{s}_1)(r_1(t) - \bar{r}_1)}{L_w} = \frac{9.766}{13} = 0.751. \tag{4}$$

Unfortunately the covariance does not tell us anything about the strength of the relationship because it is affected by the changes in the units of measurement. In case of our weather data the covariance depends on whether the temperature is measured in degree Celsius or e.g. in degree Fahrenheit. The covariance for measurements in degree Fahrenheit would be greater than the covariance for measurements in degree Celsius. To avoid this disadvantage, we *standardize* the data first, by dividing the covariance with the standard deviations of r_1 and s_1 which is defined for r_i as

$$\sigma(r) = \sqrt{\frac{\sum_{t \in W}(r(t) - \bar{r})^2}{L_w}} \tag{5}$$

and for s_i respectively. Using the Equation 3 and 5 resulting in:

$$PCC(s,r) = \frac{Cov(s,r)}{\sigma(r) \cdot \sigma(s)} = \frac{\sum_{t \in W}(s(t) - \bar{s})(r(t) - \bar{r})}{\sqrt{\sum_{t \in W}(r(t) - \bar{r})^2}\sqrt{\sum_{t \in W}(s(t) - \bar{s})^2}} \tag{6}$$

which is equivalent to Equation 1. Using the quantities in Table 4, the standard deviations for our data set are

$$\sigma(r_1) = \sqrt{\frac{\sum_{t \in W}(r_1(t) - \bar{r}_1)^2}{L_w}} = \sqrt{\frac{15.637}{13}} = 0.801 \text{ and} \tag{7}$$

$$\sigma(s_1) = \sqrt{\frac{\sum_{t \in W}(s_1(t) - \bar{s}_1)^2}{L_w}} = \sqrt{\frac{8.348}{13}} = 1.097. \tag{8}$$

Using the results of Equation 4, 7 and 8 in Equation 6 we obtain

$$\begin{aligned} PCC(s_1,r_1) &= \frac{\sum_{t \in W}(s_1(t) - \bar{s}_1)(r_1(t) - \bar{r}_1)}{\sqrt{\sum_{t \in W}(r_1(t) - \bar{r}_1)^2}\sqrt{\sum_{t \in W}(s_1(t) - \bar{s}_1)^2}} \\ &= \frac{0.751}{2.889 \cdot 3.954} = 0.855. \end{aligned} \tag{9}$$

The high value of $PCC(s_1, r_1) = 0.855$ is consistent with the strong linear relationship between s_1 and r_1 we saw in the scatter plot (Figure 3). If PCC equals 1 (-1), the positive (negative) linear relationship between s_1 and r_1 would be perfect. In this case all time points would lie on one line, the so called regression line. If PCC equals zero, no linear relationship exists. In order to show how the time point's distribution relates to the regression line and the PCC, we introduce regression analysis. *Regression analysis* is an extension to correlation analysis because it postulates a model that can be used not only to measure the direction and the strength of a relationship between the reference time series and the base time series but also to numerically describe that relationship.

4.1 Linear Regression

In linear regression the relationship between a reference time series r and its base time series s is postulated as a linear model

$$s(t) = \beta_0 + \beta_1 r(t) + \epsilon_t \tag{10}$$

where β_0 and β_1 are constants called the *model regression coefficients* and ϵ is a random error [2]. It is assumed that the linear model (Equation 10) provides an acceptable approximation to the true relationship between r and s. In

$t \in W$	$(s_1(t) - \bar{s}_1)^2$	$(r_1(t) - \bar{r}_1)^2$
1	2.296	2.462
2	1.244	0.756
3	1.730	0.592
4	1.031	0.136
5	2.609	0.072
6	0.235	0.220
7	0.007	0.072
8	0.342	0.053
9	1.176	0.282
10	1.650	0.866
11	0.783	1.279
12	1.917	0.866
13	0.616	0.690
Sum	15.637	8.348

Table 4: Quantities needed to compute the standard deviations of the temperature series of Visp (s_1) and Siders (r_1).

other words we assume a linear relationship between the time series r_1 and s_1 of our weather data shown in Table 2. A linear relationship between r_1 and s_1 exists since we obtain a strong positive correlation of $PCC(s_1, r_1) = 0.855$ and we see a linear trend in the scatter plot (Figure 4).

The regression line is the best-fitting line using the *least squares method* [2]. The coefficient β_1 in Equation 10, called the *slope*, may be interpreted as the change in s for one unit change in r. The constant coefficient β_0, called the *intercept*, is the predicted value of $s(t)$ when $r(t) = 0$.

Based on the available data in Table 2, we wish to estimate the parameters β_0 and β_1. This is equivalent to finding the straight line that gives the best representation of the points in the scatter plot of the times series s_1 versus the time series r_1 (Figure 4). We estimate the parameters using the mentioned least squares method, which gives the line that minimizes the sum of squares of the *vertical distances* (Figure 5) from each point to the line. We square the vertical distances in order to

- get only positive distances, such that positive and negative distances do not cancel out,
- emphasize larger differences.

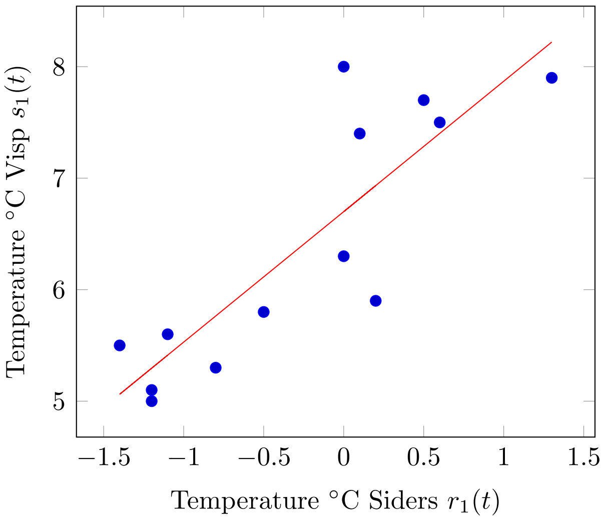

Figure 4: Data set's (Table 2) scatter plot indicating the regression line.

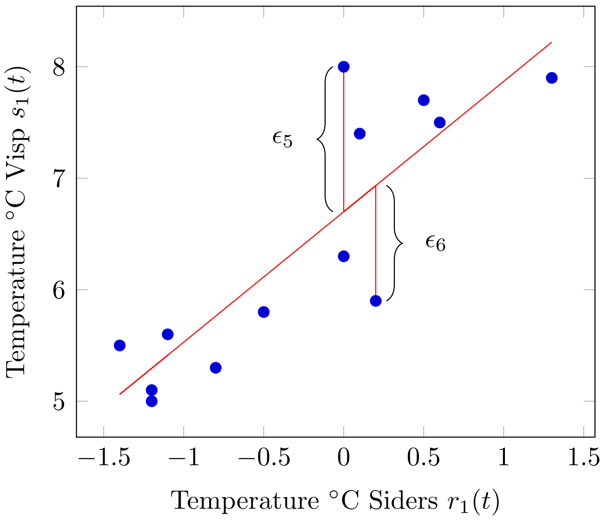

Figure 5: Data set's (Table 2) scatter plot indicating the regression line and the error distances ϵ_5 and ϵ_6.

The vertical distances represent the errors in s_1. These errors can be obtained by rewriting Equation 10 as

$$\epsilon_t = s(t) - \beta_0 - \beta_1 r(t). \tag{11}$$

The sum of squares of these distances can be written as

$$Dist(\beta_0, \beta_1) = \sum_{t \in W} \epsilon_t^2 = \sum_{t \in W} (s(t) - \beta_0 - \beta_1 r(t))^2. \tag{12}$$

We find the values β_0, β_1 that minimize $Dist(\beta_0, \beta_1)$ in order to get the regression line that best approximates the data set. The first derivative of Equation 12 provides us the values of β_0 and β_1 that minimize $Dist(\beta_0, \beta_1)$. We partially differentiate with respect to β_0 and β_1 and solve for the critical points:

$$\frac{\partial Dist(\beta_0, \beta_1)}{\partial \beta_0} = -2 \sum_{t \in W} (s(t) - \beta_0 - \beta_1 r(t) = 0 \tag{13}$$

$$\frac{\partial Dist(\beta_0, \beta_1)}{\partial \beta_1} = -2 \sum_{t \in W} (s(t) - \beta_0 - \beta_1 r(t))r(t) = 0. \tag{14}$$

Solving for β_0 and β_1 we obtain (*see Appendix A for details*):

$$\beta_0 = \bar{s} - \beta_1 \bar{r} \tag{15}$$

$$\beta_1 = \frac{\sum_{t \in W} (s(t) - \bar{s})(r(t) - \bar{r})}{\sum_{t \in W} (r(t) - \bar{r})^2}. \tag{16}$$

Once β_1 is calculated, then we can obtain β_0 by using Equation 15.
β_0 and β_1, called the least squares estimates, denote the intercept and the slope of the line that has the smallest possible sum of squares of the vertical distances from each point to the line. For this reason the line is called *least squares regression line* and it is given by

$$\hat{s}(t) = \beta_0 + \beta_1 r(t). \tag{17}$$

Using Table 3 and 4 we obtain for our data set

$$\beta_1 = \frac{\sum_{t \in W} (s_1(t) - \bar{s}_1)(r_1(t) - \bar{r}_1)}{\sum_{t \in W} (r_1(t) - \bar{r}_1)^2} = \frac{9.766}{8.348} = 1.170 \tag{18}$$

and

$$\beta_0 = \bar{s}_1 - \beta_1 \bar{r}_1 = 6.385 - 1.170 \cdot (-0.269) = 6.700. \tag{19}$$

Thus the regression line intersects the y-axis $\hat{s}_1(t) = 6.700$ when $r_1(t) = 0$ and if the temperature in Siders increases one degree Celsius, the temperature in Visp increases 1.170 degrees Celsius. The Equation of the regression line in Figure 4 then results using Equation 17 in

$$\hat{s}_1(t) = 6.700 + 1.170 \cdot r_1(t). \tag{20}$$

The least squares line always exists because we can always find a line that gives the minimum sum of squares of the vertical distances. The t-th fitted value $\hat{s}(t)$ represents the point on the least squares regression line (Equation 17) corresponding to $r(t)$. The vertical distance corresponding to the t-th observation is

$$\epsilon_t = s(t) - \hat{s}(t). \tag{21}$$

These vertical distances are called the *ordinary least squares residuals*. One property of the residuals in Equation 21 is that their sum is zero. This means that the sum of the vertical distances above the regression line is equal to the sum of the vertical distances below (*see Appendix B for details*):

$$\sum_{t \in W} \epsilon_t = 0. \tag{22}$$

Using Equation 19 and 21 we are now able to calculate the fitted values $\hat{s}_i(t)$ and the ordinary least squares residuals ϵ_t as we are now in the possession of the least squares estimates β_0 and β_1. E.g. for data point $t = 5$ they result in

$$\hat{s}_1(5) = 6.700 + 1.170 \cdot 0 = 6.700, \tag{23}$$

and

$$\epsilon_5 = 8 - 6.700 = 1.300. \tag{24}$$

Study Figure 5 to see $\hat{s}_1(t)$ and ϵ_t for the points $(r(5), s(5))$ and $(r(6), s(6))$ in the scatter plot referencing Table 5 for the fitted values $\hat{s}_1(t)$ and the ordinary least squares residuals ϵ_t of all weather data.
The closer the data points scatter around the regression line in Figure 4, the closer is the PCC to ±1, as we will show next.

4.2 Measuring the Quality of Fit

After fitting a linear model relating s to r, we are not only interested in knowing whether a linear relationship exists and in predicting or imputing values but also in measuring the quality of the fit of the model to the data. Therefore we introduce a useful measure called the *coefficient of determination* denoted by R^2 [2]. R^2 can be interpreted as the proportion of the total

t	$\hat{s}_1(t)$	ϵ_t
1	8.220	-0.320
2	7.402	0.098
3	7.285	0.415
4	6.817	0.583
5	6.700	1.300
6	6.934	-1.034
7	6.700	-0.400
8	6.115	-0.315
9	5.764	-0.464
10	5.296	-0.196
11	5.062	0.438
12	5.296	-0.296
13	5.413	0.187
Sum	83.000	0.000

Table 5: The fitted values $\hat{s}_1(t)$ and the ordinary least squares residuals ϵ_t for the weather data.

variation in s that is accounted for by the predictor r. It is the square of the already introduced PCC, such that

$$[PCC(s,r)]^2 = R^2. \tag{25}$$

The coefficient of determination is developed as follows: After we compute the least squares estimates of the parameters of a linear model, let us compute the following quantities:

$$\begin{aligned} SST &= \sum_{t \in W} (s(t) - \bar{s})^2, \\ SSR &= \sum_{t \in W} (\hat{s}(t) - \bar{s})^2, \\ SSE &= \sum_{t \in W} (s(t) - \hat{s}(t))^2 = \sum_{t \in W} \epsilon_t^2, \end{aligned} \tag{26}$$

where SST stands for the total sum of squared deviations of s from its mean $\bar{s}$, SSR denotes the sum of squares due to regression, and SSE represents the sum of squared residuals (vertical error distances). From [2] we know that:

$$R^2 = \frac{SSR}{SST}. \tag{27}$$

The quantities $(s(t)-\bar{s})$, $(\hat{s}(t)-\bar{s})$ and $(s(t)-\hat{s}(t))$ are presented in Figure 6 for a typical point $(r(t), s(t))$. A horizontal line is drawn at $s(t)=\bar{s}$. Note that for every point (s, r), there are two points, $(r(t), \hat{s}(t))$, which lies on the fitted line, and $(r(t), \bar{s}(t))$ which lies on the line $s(t)=\bar{s}$. According to Figure 6, the total sum of squared deviations, SST, in s can be decomposed into the sum of two quantities. The first, SSR, measures the quality of r as a predictor of s. The second, SSE, measures the error in this prediction. Next we want to combine the quantities SST, SSR and SSE in a single equation, as this will allow us to prove that if all points in the scatter plot lie on the regression line, the PCC reaches its maximum. By definition we have $s(t)=\hat{s}(t)+\epsilon_t$. Subtracting $\bar{s}$ on both sides of this expression, we have

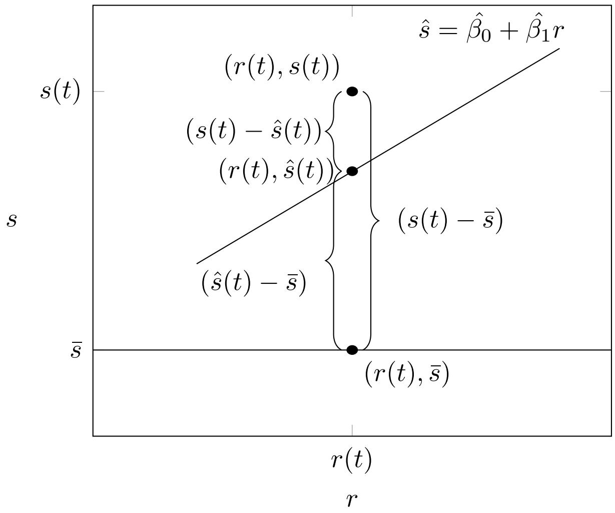

Figure 6: A graphical illustration of quantities defined in Equation 26, shown

$$s(t)-\bar{s}=\hat{s}(t)-\bar{s}+\epsilon_t.$$

Squaring both sides:

$$\begin{aligned}(s(t)-\bar{s})^2 &= ((\hat{s}(t)-\bar{s})+\epsilon_t)^2\\ &= (\hat{s}(t)-\bar{s})^2+\epsilon_t^2+2\epsilon_t(\hat{s}(t)-\bar{s}).\end{aligned}$$

Summing for all t:

$$\sum_{t\in W}(s(t)-\bar{s})^2=\sum_{t\in W}(\hat{s}(t)-\bar{s})^2+\sum_{t\in W}\epsilon_t^2+2\sum_{t\in W}\epsilon_t(\hat{s}(t)-\bar{s}). \quad (28)$$

It can be shown that the last term on the right hand side, i.e. $2\sum_{t\in W}\epsilon_t(\hat{s}(t)-\bar{s})$, equals zero (*see Appendix C for details*). Hence Equation 28 simplifies to

$$\sum_{t\in W}(s(t)-\bar{s})^2=\sum_{t\in W}(\hat{s}(t)-\bar{s})^2+\sum_{t\in W}\epsilon_t^2. \quad (29)$$

Thus the equality is given by

$$SST = SSR + SSE. \tag{30}$$

Using Equation 25 and 27 we have

$$[PCC(s,r)]^2 = R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}. \tag{31}$$

Note that $0 \leq R^2 \leq 1$ because $SSE \leq SST$. If R^2 is near 1, then r accounts for a large part of the variation in s. For this reason, R^2 is known as the coefficient of determination because it gives us an idea of how strongly r determines s. Considering Equation 31, the smaller the vertical error distances SSE, the greater R^2 and PCC accordingly and vice versa.

Theorem 1. *Let r and s be two time series. Let $\hat{s}(t) = \beta_0 + \beta_1 r(t)$ be the least squares regression line. $|PCC(s,r)| = 1$ if and only if there is a perfect linear relationship between r and s, i.e. $\forall\ t \in W(s(t) = \hat{s}(t))$.*

Proof. First observe that $\forall\ t \in W(s(t) = \hat{s}(t))$ is equivalent to $SSE = \sum_{t \in W}(s(t) - \hat{s}(t))^2 = 0$.
(If-part) We assume $SSE = 0$, hence $PCC(s,r)^2 = 1 - \frac{0}{SST} = 1$ and $|PCC(s,r)| = 1$.
(Only-if-part) We assume $|PCC(s,r)| = 1$, hence $1^2 = 1 - \frac{SSE}{SST}$, which yields $\frac{SSE}{SST} = 0$. Consequently $SSE = 0$. □

Using our data set defined in Table 2 and the fitted values in Table 5 as well as the defined PCC in Equation 6 we can calculate $PCC(s_1, r_1) = 0.855$ from which it follows that $R^2 = 0.855^2 = 0.731$. The same value can be computed using Equation 27 and the quantities in Table 6:

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{4.211}{15.637} = 0.731. \tag{32}$$

The value of $R^2 = 0.731$ indicates that 73.1% of the total variability in s_1 is accounted for by r_1. It also indicates that a strong linear relationship between r_1 and s_1 exists.

t	$(s_1(t) - \bar{s}_1)^2$	$(\hat{s}_1(t) - \bar{s}_1)^2$	$(s_1(t) - \hat{s}_1(t))^2$
1	2.296	3.370	0.103
2	1.244	1.034	0.010
3	1.730	0.810	0.173
4	1.031	0.187	0.340
5	2.609	0.099	1.691
6	0.235	0.301	1.068
7	0.007	0.099	0.160
8	0.342	0.073	0.099
9	1.176	0.386	0.215
10	1.650	1.186	0.038
11	0.783	1.750	0.192
12	1.917	1.186	0.087
13	0.616	0.945	0.035
Sum	15.637	11.426	4.211

Table 6: The quantities SST, SSR and SSE for the weather data.

5 Case Matching Similarity

As shown the PCC can be used to measure the strength of linear correlation between time series. The Case Matching Similarity (CMS) can also discover the existence of non-linear correlations between time series. A time series value in a non-linearly correlated data set does not change proportionally as changing the other time series's value. However non-linearity does not mean time series can not correlate.

Lets consider the two time series from Figure 7. The red line represents a base time series s_2, where $s_2(t) = \sin^2(t)$. The blue line represents a reference time series r_2, where $r_2(t) = \sin(t)$. The scatter plot in Figure 8 clearly shows that between the time series r_2 and s_2 exists no linear relationship. However, they are still related, since $s_2(t) = r_2(t)^2$, i.e. time series s_2 can be seen as a function of r_2.

CMS considers this fact and intuitively checks if similar values of any two time series s and r co-occur frequently. A time series s is considered similar to r if for any two time points $t_1, t_2 \in W$, s has similar values at t_1 and t_2 if r has similar values at these two time points. Therefore, on a high level, CMS tries to measure how similar the value $s(t_1)$ and $s(t_2)$ are to each other when $r(t_1) \approx r(t_2)$.

The Case Matching Similarity was developed in the context of the impu-

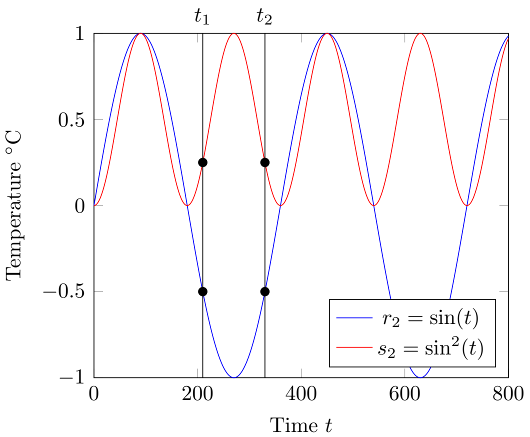

Figure 7: Example of two non-linearly correlated time series r_2 and s_2.

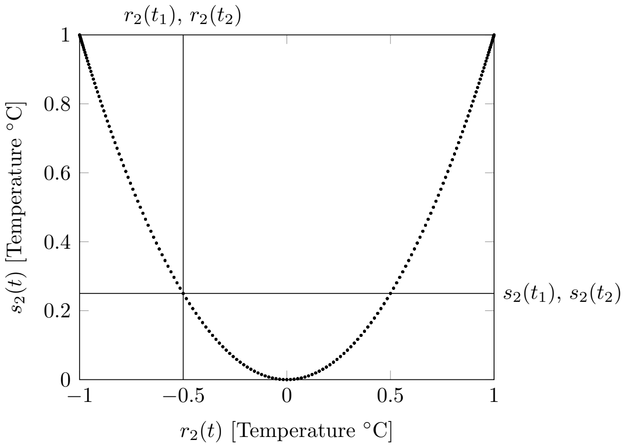

Figure 8: The scatterplot of r_2 and s_2 shows the non-linear correlation of two time series.

tation algorithm TKCM [1]. A missing value $s(t_2)$ in a base time series s is imputed using one or more reference time series r. TKCM, in its simplest form, looks for values $r(t_1)$ similar to $r(t_2)$ and uses the corresponding value $s(t_1)$ of the base time series s to impute the missing value $s(t_2)$. TKCM assumes that whenever the reference time series observes similar values, also the base time series does and CMS is designed to find reference time series that satisfy this property.

Example 5.1. *To show an example of co-occurrence, let us consider Figure 7. Assuming $t_1 = 210$ and $t_2 = 330$, we have that $r_2(t_1) = r_2(t_2) = -0.5$ and $s_2(t_1) = s_2(t_2) = 0.25$. From the scatter plot in Figure 8, we conclude that for every value of $r_2(t)$, time series s_2 has exactly one value $s_2(t)$. Due to the relation of s_2 and r_2 illustrated in Figure 8, time series r_2 can be used to impute a missing value in s_2 using TKCM [1]. Intuitively, if $s(t_2)$ is missing,TKCM looks for values similar to $r(t_2) = -0.5$ and would find $r(t_1) = -0.5$. The value $s(t_1) = 0.25$ is used to impute $s(t_2)$, which would in this case be the correct imputation.*

CMS proposes an algorithm to actually measure the strength of co-occurrence between two time series s and r, where a small CMS implies a strong co-occurrence and a large CMS a weak co-occurrence. CMS splits the range of time series r into equal-sized sub-ranges, called buckets. Each bucket b_z contains the values of s such that the corresponding value of r is within the bucket limits.

$$b_z = \{s(t) | t \in W \wedge zw \leq r(t) < (z+1)w\} \tag{33}$$

b_z	t	$r_1(t)$	$s_1(t)$
b_{-4}	01:20	-1.4	5.5
b_{-3}	01:00	-1.2	5.1
	01:40	-1.2	5
	02:00	-1.1	5.6
b_{-2}	00:20	-0.5	5.8
	00:40	-0.8	5.3
b_{-0}	23:00	-0.1	7.4
	23:20	0	8
	23:40	0.2	5.9
	00:00	0	6.3
b_1	22:20	0.6	7.5
	22:40	0.5	7.7
b_3	22:00	1.3	7.9

Table 7: Buckets b_z with corresponding $s_1(t)$ for bucket width $w = 0.4$.

with $z \in \mathbb{Z}$ as the bucket's ID and $w \in \mathbb{R}_{>0}$ as the bucket's width.

Example 5.2. *We compute the buckets for our running example data from Table 2 with bucket width $w = 0.4$. Using our weather data from Table 2, each value $s_1(t)$ gets mapped into the corresponding bucket b_z. In Table 7 the buckets b_z, timestamps t, the values from reference time series $r_1(t)$ and the corresponding values $s_1(t)$ are shown. For example, bucket $b_{-3} = \{s_1(\mathit{01{:}00}), s_1(\mathit{01{:}40}), s_1(\mathit{02{:}00})\}$ contains all values $s_1(t)$ such that $-3\times 0.4 = -1.2 \leq r_1(t) < -0.8 = -2 \times 0.4$.*

CMS then calculates each bucket's standard deviation. The smaller a bucket's standard deviation is, the closer are the values $s(t) \in b_z$ to the bucket mean $\bar{b_z}$, the more similar are they to each other and the stronger is the co-occurrence. Formally, the standard deviation for each bucket is defined as

$$\sigma_z = \sqrt{\frac{1}{|b_z|} \sum_{s(t) \in b_z} (s(t) - \bar{b_z})^2} \tag{34}$$

with $\bar{b_z} = \frac{1}{|b_z|} \sum_{s(t) \in b_z} s(t)$ as the bucket's mean.

Let $B = \{b_z | b_z \neq \emptyset\}$ be the set of all non-empty buckets. Then CMS is defined as the average bucket standard deviation, where each term is weighted by the number of elements in the corresponding bucket

$$CMS_w(s, r) = \frac{1}{|B|} \sum_{b_z \in B} \frac{|b_z|}{L_W} \sigma_z \tag{35}$$

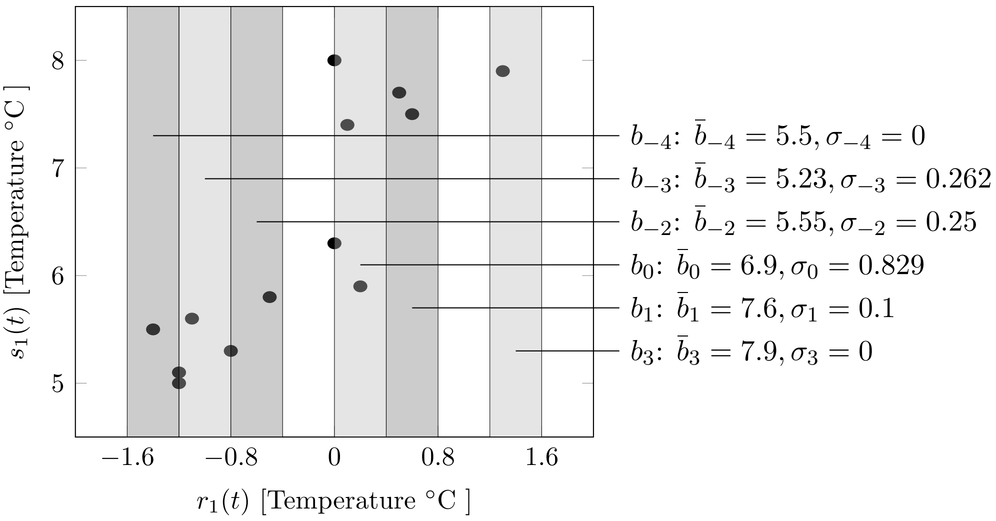

Figure 9: A graphical illustration of buckets in the scatter plot of r_1 and s_1.

The reason for the weighting factor $\frac{|b_z|}{L_W}$ of the bucket's standard deviation σ_z is that a bucket that contains many values should have a bigger influence on the CMS than a bucket that contains only few values. In our example, the standard deviation from bucket b_1 will be multiplied by $\frac{2}{13}$, and therefore the weighted standard deviation σ_1 is 0.015.

Example 5.3. *Continuing with our buckets from Table 7, we calculate the the standard deviation σ_z for each bucket. The standard deviation indicates how much the bucket's values differ from its average. Figure 9 shows the scatter plot between r_1 and s_1, where the buckets partition the x-axis. For each bucket we indicate the bucket mean $\bar{b_z}$ and standard deviation σ_z.*

Finally CMS calculates the sum of the average bucket standard deviation weighted by the number of elements in the corresponding bucket. For $w = 0.4$ CMS is defined as $CMS_{0.4}(s_1, r_1) = \frac{1}{6} \times (\frac{1}{13} \times 0 + \frac{3}{13} \times 0.262 + \frac{2}{13} \times 0.25 + \frac{4}{13} \times 0.839 + \frac{2}{13} \times 0.1 + \frac{1}{13} \times 0) = 0.0621$.

5.1 Asymmetry

As we have seen in Section 4, the Pearson Correlation Coefficient is symmetric, which means that $PCC(s, r) = PCC(r, s)$. This does no apply for Case Matching Similarity, where in general $CMS_w(s, r) \neq CMS_w(r, s)$. Figure 11 illustrates the scatter plot of $CMS_w(r_2, s_2)$, where again $r_2(t) = \sin(t)$ and $s_2(t) = \sin^2(t)$. Compared to Figure 8, which presents the scatter plot of $CMS_w(s_2, r_2)$, the shape of the data points in Figure 11 has rotated 90 degrees to left. Observe that whenever time series s_2 has some value $s_2(t)$, time series r_2 can have two different values, except when $s_2(t) = 0$, in which case $r_2(t) = 0$ too. For example consider the time points $t_2 = 330$ and $t_3 = 510$. We have that $s_2(t_2) = s_2(t_3) = 0.25$, but $r_2(t_2) = -0.5$ and $r_2(t_3) = 0.5$. Therefore, whenever s_2 has similar values, it is not the case that r_2 has similar values too. For this reason, we expect $CMS_w(s_2, r_2) < CMS_w(r_2, s_2)$ and indeed this is the case as $CMS_w(s_2, r_2) = 0.01 < CMS_w(r_2, s_2) = 0.13$ for $w = 0.25$. The asymmetry of CMS is a positive property. As mentioned above, CMS is used to find the most suitable reference time series r for a base time series s. In reverse, s can be a bad reference time series for r. This example has shown that using s as a reference time series to impute a missing value in r with TKCM can yield very bad results, as TKCM would find widely varying values of r (e.g 0.5 and -0.5) and could not distinguish between good and bad candidates.

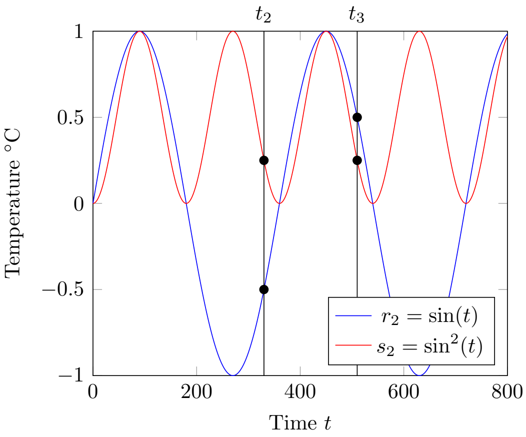

Figure 10: Two non-linearly correlated time series r_2 and s_2.

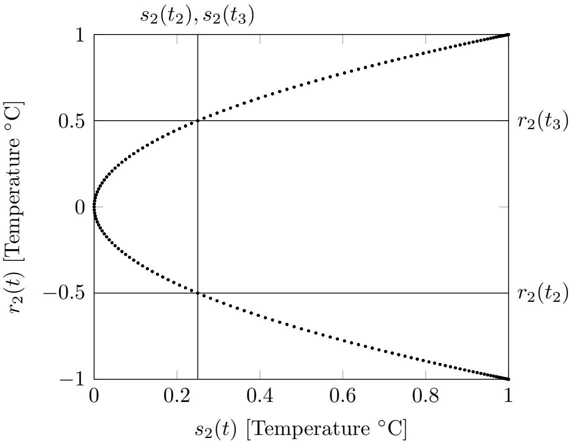

Figure 11: The scatterplot of s_2 and r_2.

5.2 Analysis

$CMS_w(s, r)$ ranges from 0, in which case s and r perfectly correlate, to ∞ (excluded). In this subsection, we show when CMS reaches its minimum.

Theorem 2. *Let r and s be two time series over the sliding window W. CMS reaches its minimum, zero, as bucket width w approaches zero if and only if s has the same value for any two time points $t_1, t_2 \in W$ when r has the same value for t_1, t_2, i.e.*

$$\lim_{w \to 0} CMS_w(s, r) = 0 \iff \forall t_1, t_2 \in W : r(t_1) = r(t_2) \to s(t_1) = s(t_2)$$

Proof. (If-Part) We assume $\forall t_1, t_2 \in W : r(t_1) = r(t_2) \rightarrow s(t_1) = s(t_2)$. Observe that for any two values $s(t_1), s(t_2) \in b_z$ that belong to the same bucket b_z, by definition we have that $|r(t_1) - r(t_2)| \leq w$. Hence, for w approaching 0, we have that $|r(t_1) - r(t_2)| \leq 0$, implying that $r(t_1) = r(t_2)$. By assumption we know that if $r(t_1) = r(t_2)$ we have that $s(t_1) = s(t_2)$, hence for any two values $s(t_1), s(t_2) \in b_z$ in a bucket b_z we know that $s(t_1) = s(t_2) = \bar{b_z}$. Consequently, every standard deviation $\sigma_z = 0$ and $\lim_{w \to 0} CMS_w(s, r) = 0$.

(Only-if-part) We assume $\lim_{w \to 0} CMS_w(s, r) = 0$. Since $w \rightarrow 0$, we know that for any two time points t_1, t_2 the values $s(t_1), s(t_2) \in b_z$ are in the same bucket only if $r(t_1) = r(t_2)$. Moreover, since $\lim_{w \to 0} CMS_w(s, r) = 0$ we know that $\sigma_z = 0$ for any bucket $b_z \in B$. The bucket standard deviation σ_z can only be 0 if every value $s(t_1), s(t_2) \in b_z$ is equal. Combining these two facts we know that $\forall t_1, t_2 \in W : r(t_1) = r(t_2) \rightarrow s(t_1) = s(t_2)$. □

To interpret Theorem 2, we use the examples from Figure 8 and 11. Imagine the bucket width w approaches zero, every distinct measurement $s_2(t)$ in Figure 8 would be in a separate bucket and the standard deviation of each bucket b_z will be $\sigma_z = 0$, and therefore, $\lim_{w \to 0} CMS_w(s_2, r_2) = 0$. Considering Figure 11, with bucket width w approaching zero, every bucket contains two distinct measurements $r_2(t)$ (e.g. $r_2(330) = -0.5$ and $r_2(510) = 0.5$, as before), except bucket b_0 which only contains one distinct measurement $r_2(t) = 0$. Therefore the bucket standard deviation $b_z \neq 0$ and $\lim_{w \to 0} CMS_w(r_2, s_2) \neq 0$.

5.3 Bucket Width w

The Case Matching Similarity of two time series s and r strongly depends on bucket width w. There exists a tradeoff, choosing a too small bucket width w, results that each distinct value gets its own bucket and therefore there are too many buckets. Having many buckets leads to less values within the bucket. Having only a few values in a bucket makes the bucket's standard deviation less meaningful and less representative. Choosing a too large bucket width w results that all values are put in one bucket and therefore too few buckets. Choosing a too big bucket width results in a higher distribution of values within the bucket which increases the standard deviation and therefore lead to a higher CMS. To illustrate the effect of the bucket width we ran an experiment with our sample time series s_1 and r_1. The graph in Figure 12 shows the heavy impact of the bucket width w on the CMS, as the range of $CMS_w(s_1, r_1)$ increases from 0.012 for $w = 0.05$ till 0.267 for $w = 2.5$. It is expected that $CMS_w(s_1, r_1)$ will be never 0, since

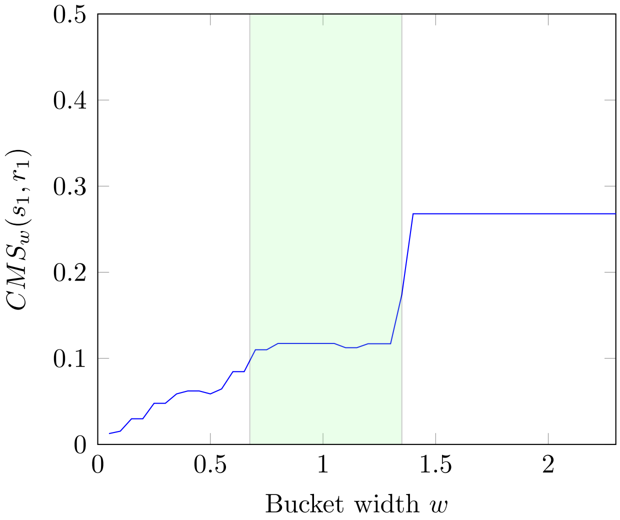

Figure 12: Effect of bucket width w on $CMS_w(s_1, r_1)$.

even with w approaching 0, there will be two buckets with more than one value. For example r_1(01:00) and r_1(01:40) have the same measured value of -1.2°C, and therefore $s_1(01{:}00) = 5.1$ and $s_1(01{:}40) = 5.0$ will be always in the same bucket and their standard deviation can never be zero. Starting with $w = 0.05$, as expected $CMS_w(s_1, r_1)$ is growing as w grows. The curve plateaus for $w \in [0.6, 1.35]$, because then there are only two relevant buckets. We call a bucket relevant if it contains more than one value and hence can have a non-zero standard deviation. With bucket width $w = 0.6$, the only two relevant buckets are $b_{-2} = \{s_1(00{:}40), s_1(01{:}00), s_1(01{:}40), s_1(02{:}00)\}$ and $b_0 = \{s_1(22{:}40),\ s_1(23{:}00), s_1(23{:}20), s_1(23{:}40), s_1(00{:}00)\}$. Starting with $w = 1.35$, $CMS_w(s_1, r_1)$ shoots up and flattens out at 0.267. With $w = 1.4$, again all values are distributed in two buckets. With $w = 0.6$ there were also only two relevant buckets, however, $CMS_{1.4}(s_1, r_1)$ has a three times higher CMS than $CMS_{0.6}(s_1, r_1)$. This is because of the weighting of the bucket's standard deviation. Since with $w = 1.4$ buckets contain more values, the standard deviation of that bucket has a higher weight. From $w = 1.4$, CMS does not change anymore, because the number of buckets does not change anymore.

A formula has been developed to determine an interval for a meaningful bucket width w. As shown above, the bucket width has an impact on the number of values m which are stored within one bucket. To calculate a statistically valid and meaningful standard deviation, at least $m_d \geq 3$ values per bucket are desired.

Consider again Figure 9, on its x-axis we have the values of reference time series r, sorted by value. As shown before, the bucket width w partitions the x-axis of the scatter plot. To ensure that a bucket is "wide" enough and contains, on average, at least m_d values, we look at the average absolute difference c between two consecutive measurements of r on the x-axis of the scatter plots. In other words, we compute the average absolute difference c between two consecutive measurements when reference time series r is sorted by value. We denote with r' time series r sorted by value. That is, $r' = \langle r(t_i), r(t_y), \dots, r(t_{L_W}) \rangle$, where $r'(t_i) \leq r'(t_y)$. Then c is given by,

$$c = \frac{1}{L_W - 1} \sum_{t_i \in W \setminus \{t_{L_W}\}} |r'(t_i) - r'(t_{i+1})|$$

Applying this formula on the reference time series r_1 in our sample data set we get $c_{r_1} = 0.225$.

To calculate a lower-bound of w, the minimum desired number of values in a bucket m_d is multiplied with the average difference between two consecutive measurements c, $w \geq m_d c$. The lower-bound of w ensures that all buckets store on average at least m_d values.

The upper-bound of w ensures that bucket width w is not too big and not all values are stored in one single bucket, i.e. we are aiming to have at least two buckets. Therefore to calculate the upper-bound of w, the difference of the maximum and minimum of r is divided by 2, i.e. $w \leq \frac{max(r(t)) - min(r(t))}{2}$.

Having calculated c and determined the desired number of values in bucket m_d, the lower- and upper-bound formulas can be merged in one formula to calculate an interval for w. Equation 36 show the formula to calculate the interval for w.

$$m_d c \leq w \leq \frac{max(r(t)) - min(r(t))}{2} \tag{36}$$

Applying this formula on our sample data set we get an interval of $[0.675, 1.35]$ for w. The green area in Figure 12 illustrates this interval.

Generally, there exists no universal value for w which is optimal for all time series since the bucket width w depends heavily on the time series characteristics. The formula from Equation 36 provides an interval for w by excluding the extremes, we aim to have on average at least m_d values in a bucket and have at least two buckets. However, it remains an open issue to find an optimal w inside this interval.

6 Implementation

Computing the Pearson Correlation Coefficient and Case Matching Similarity for a streaming time series can be CPU and memory expensive if they are recalculated from scratch whenever the streams produce a new value and the sliding window W shifts forward. To avoid that all measurements from W have to be considered for every recalculation of PCC and CMS, we propose to compute the two measures incrementally from this new stream values.

For both measures a PCC-Object *po*, respectively a CMS-Object *co* is created. Theses objects provide the necessary state that enables us to compute the two measures incrementally as well as three functions, *AddMeasurement*, *RemoveMeasurement* and *CalculatePCC*, respectively *CalculateCMS*. The objects itself will be explained below in Section 6.1 respectively 6.2.

To simulate a streaming time series, we introduce a sliding window W of size L_W which slides over the time series. The simulation is for both similarity measures identical. Algorithm 1 shows the simulation of streaming time series and the computation of CMS and PCC on this streams. The simulation consists of two phases, the growing phase and the sliding phase. In the growing phase the simulation only adds values to sliding window W using the *AddMeasurement*-Function because it does not contain L_W time points yet ($|W| < L_W$). In the sliding phase, the sliding window W is full ($|W| = L_W$) and slides over the time series, therefore every time a new value is added, a old value gets removed. As shown in Algorithm 1, with every measurement of s, the *po*-Object respectively the *co*-Object is updated through the function *AddMeasurement*. In the sliding phase, always the last measurement of the sliding window will be removed through the function *RemoveMeasurement*-Function.

Algorithm 1: Main

```
begin
    $W \leftarrow \{\}$
    for $s(t) \in s$ do
        $W \leftarrow W \cup \{t\}$
        $t_{exp} \leftarrow t - L_W$
        for $r \in \mathbf{R}$ do
            $po \leftarrow r.po$
            $co \leftarrow r.co$
            $AddMeasurement(po, s(t), r(t))$
            $AddMeasurement(co, s(t), r(t))$
            if $t_{exp} \in W$ then
                $RemoveMeasurement(po, s(t_{exp}), r(t_{exp}))$
                $RemoveMeasurement(co, s(t_{exp}), r(t_{exp}))$
            end
        end
        $W \leftarrow W \backslash \{t_{exp}\}$
    end
end
```

6.1 PCC

In Section 4, Equation 1 has been introduced to calculate the linear correlation of two time series through the PCC. For the implementation of the PCC, an alternative but equivalent formula has been used to compute the PCC. The used formula in the implementation of the PCC is represented in Equation 37 [4].

$$PCC(s,r) = \frac{n \times (\sum\limits_{t \in W} s(t) \times r(t)) - (\sum\limits_{t \in W} s(t)) \times (\sum\limits_{t \in W} r(t))}{\sqrt{[n \sum\limits_{t \in W} s(t)^2 - (\sum\limits_{t \in W} s(t))^2] \times [n \sum\limits_{t \in W} r(t)^2 - (\sum\limits_{t \in W} r(t))^2]}} \tag{37}$$

$$= \frac{n \times sum\_sr - sum\_s \times sum\_r}{\sqrt{[n \times sum\_s^2 - (sum\_s)^2] \times [n \times sum\_r^2 - (sum\_r)^2]}} \tag{38}$$

This alternative formula in Equation 37 can be split into independent components which makes it trivial to calculate the PCC. Examining Equation 37, we see that the equation consists of six terms.

1. n, the number of measurements, $n = |W|$
2. $\sum s(t) \times r(t)$, the sum of the product of $s(t) \times r(t)$

3. $\sum s(t)$, the sum of all values from s
4. $\sum r(t)$, the sum of all values from r
5. $\sum s(t)^2$, the sum of the square of the measurement $s(t)$
6. $\sum r(t)^2$, the sum of the square of the measurement $r(t)$

All these six terms can be trivially calculated incrementally. As already mentioned above, we use a *po*-Object to keep track of these components. The *po*-Object has therefore the following structure $\langle n$, sum_sr, sum_s, sum_r, sum_s^2, sum_r^2, $PCC\rangle$. Compared to Equation 37, the *po*-Object has seven elements. The seventh element PCC, is the return value of Equation 37. Rewriting the components from Equation 37 with the elements used in the *po*-Object, we get Equation 38.

When the sliding window W slides forward, the $Main$-Function calls the *AddMeasurement*-Function in Algorithm 2 to add a new value pair, a measurement of $s(t)$ and $r(t)$. *AddMeasurement*-Function then updates the PCC-Object *po* and calls the *CalculatePCC*-Function in Algorithm 4. *CalculatePCC* then calculates the PCC according to the Equation 37, respectively Equation 38.

Algorithm 2: PCC *AddMeasurement*

```
Input: $po$, Measurement $s(t)$, Measurement $r(t)$
begin
    $po.sum\_s \leftarrow po.sum\_s + s(t)$
    $po.sum\_r \leftarrow po.sum\_r + r(t)$
    $po.sum\_sr \leftarrow po.sum\_sr + (s(t) \times r(t))$
    $po.sum\_s^2 \leftarrow po.sum\_s^2 + s(t)^2$
    $po.sum\_r^2 \leftarrow po.sum\_r^2 + r(t)^2$
    $po.n \leftarrow po.n + 1$
    $CalculatePCC(po)$
end
```

When the sliding window advances and W is full, the oldest timestamp t_{exp} drops out. When timestamp t_{exp} is no element anymore of W, $s(t_{exp})$ and $r(t_{exp})$ have to be removed from the PCC. The *RemoveMeasurement*-Function in Algorithm 3 updates the PCC-Object *po* through subtracting from the elements of the PCC-Object *po*. Once the PCC-Object *po* is updated, the PCC is re-calculated with the *CalculatePCC*-Function in Algorithm 4.

Algorithm 3: PCC *RemoveMeasurement*

```
Input: $po$, Measurement $s(t)$, Measurement $r(t)$
begin
    $po.sum\_s \leftarrow po.sum\_s - s(t)$
    $po.sum\_r \leftarrow po.sum\_r - r(t)$
    $po.sum\_sr \leftarrow po.sum\_sr - (s(t) \times r(t))$
    $po.sum\_s^2 \leftarrow po.sum\_s^2 - s(t)^2$
    $po.sum\_r^2 \leftarrow po.sum\_r^2 - r(t)^2$
    $po.n \leftarrow po.n - 1$
    $CalculatePCC(po)$
end
```

Algorithm 4 shows how the PCC is calculated according to Equation 38. If $n = 0$, i.e. there are no measurements in W, PCC is set to 0 to avoid a division by zero.

Algorithm 4: *CalculatePCC*

```
Input: $po$
begin
    if $po.n > 0$ then
        $po.pcc = \frac{po.n \times po.sum\_sr - po.sum\_s \times po.sum\_r}{\sqrt{[po.n \times po.sum\_s^2 - (po.sum\_s)^2] \times [po.n \times po.sum\_r^2 - (po.sum\_r)^2]}}$
    end
end
```

6.2 CMS

Similar to the calculation of the PCC, the CMS is also calculated incrementally. The CMS-Object *co* has therefore the following structure $\langle buckets[]$, w, n, $CMS\rangle$. $buckets[]$ represents a hash table of buckets, w is the specified bucket width and n is the number of measurements added to the object. From Equation 35 in Section 5, we know that the CMS is calculated from the bucket's standard deviation. To calculate the CMS incrementally, we therefore have to keep track of all buckets and calculate the bucket's standard deviation incrementally. However, we do not have to actually store measurements $s(t)$ in the buckets as long as we still can compute their standard deviation. To calculate the standard deviation, the variance is needed. Equation 39 shows how to calculate the "unnormalized variance" S_m incrementally and recursively [5].

$$S_m = S_{m-1} + (s(t) - \bar{b}_{m-1})(s(t) - \bar{b}_m), \quad \text{where } S_0 = 0 \tag{39}$$

Examining Equation 39, we see that it consists of three components besides value $s(t)$ that is added to the CMS.

1. S_{m-1}, the "unnormalized variance" before the m-th value is added to the bucket

2. $\bar{b}_{m-1}$, the mean before the m-th value is added to the bucket

3. $\bar{b}_m$, the mean after the m-th value is added to the bucket

To derive from "unnormalized variance" S_m to standard deviation σ we can calculate $\sigma = \sqrt{\frac{S_m}{m}}$. For computing mean $\bar{b}_m$ incrementally according to Equation 40, we also need the number of values m in the bucket.

$$\bar{b}_m = \bar{b}_{m-1} + \frac{1}{m}(s(t) - \bar{b}_{m-1}), \quad \text{where } \bar{b}_0 = 0 \tag{40}$$

Equation 39 and 40 are numerically stable because it avoids accumulating large sums [5]. To summarize, to calculate the standard deviation, three quantities are needed: number of values in the bucket m, current mean in the bucket $\bar{b}_m$ and the "unnormalized variance" S_m. As already mentioned above, we use a CMS-Object *co* to keep track of these components. Every bucket b_z stored in the hash table contains the three components to incrementally compute the bucket's standard deviation. It is structured as $b_z = \langle m, \bar{b}, S \rangle$.

Summarized, CMS is incremental calculated in three steps:

1. In the first step $\bar{b}_m$ is calculated using Equation 40

2. In the second step S_m is calculated using Equation 39

3. In the third step σ is calculated:

$$\sigma = \sqrt{\frac{S_m}{m}}$$

When the sliding window W shifts forward, the $Main$-Function calls the *AddMeasurement*-Function in Algorithm 5 to add a new value pair to the CMS. To look up the bucket in the hash table, first the bucket ID z is derived from measurement $r(t)$ and bucket width w. Second, a lookup is made to find the bucket b in the *buckets* hash table of the CMS-Object *co*. If b is not found a new bucket b is created and added to the hash table.

Algorithm 5: CMS *AddMeasurement*

```
Input: $co$, Measurement $s(t)$, Measurement $r(t)$
begin
    $z \leftarrow \lfloor \frac{r(t)}{co.w} \rfloor$
    $b \leftarrow HASH\_FIND(co.buckets, z)$
    if $b = NULL$ then
        $b \leftarrow \langle 0, 0, 0 \rangle$
        $HASH\_ADD(co.buckets, z, b)$
    end
    $b.m \leftarrow b.m + 1$
    $\bar{b}_{m-1} \leftarrow b.\bar{b}$
    $b.\bar{b} \leftarrow \bar{b}_{m-1} + \frac{1}{b.m}(s(t) - \bar{b}_{m-1})$
    $S_{m-1} \leftarrow b.S$
    $b.S \leftarrow S_{m-1} + (s(t) - \bar{b}_{m-1}) \times (s(t) - b.\bar{b})$
    $co.n \leftarrow co.n + 1;$
    $CalculateCMS(co)$
end
```

Once bucket b is determined, the CMS-Object co is in Algorithm 5 updated according to the step one and two of the three steps above. At the end, the *CalculateCMS*-Function in Algorithm 7 is called to calculate the final CMS.

When value pairs drops out of the sliding window W, they also get removed from the CMS through Algorithm 6. The *RemoveMeasurement*-Function works similar than the *AddMeasurement*-Function in Algorithm 5. The only difference is that it subtracts instead of adding. Therefore, for subtracting, Equation 39 changes to $S_m = S_{m+1} - (s(t) - \bar{b}_{m+1})(s(t) - \bar{b}_m)$ and Equation 40 changes to $\bar{b}_m = \bar{b}_{m+1} + \frac{1}{m}(s(t) - \bar{b}_{m+1})$. In addition, when the last value of a bucket gets removed, *RemoveMeasurement*-Function removes the bucket from the hash table.

Similar to the *AddMeasurement*-Function, the *RemoveMeasurement* -Function then calls the *CalculateCMS*-Function in Algorithm 7. The *CalculateCMS*-Function then deduces the standard deviation from the bucket's "unnormalized variance" and sums up the weighted standard deviation. Finally, CMS is calculated by dividing the sum through the number of buckets.

Algorithm 6: CMS *RemoveMeasurement*

```
Input: $co$, Measurement $s(t)$, Measurement $r(t)$
begin
    $z \leftarrow \lfloor \frac{r(t)}{co.w} \rfloor$
    $b \leftarrow HASH\_FIND(co.buckets, z)$
    $b.m \leftarrow b.m - 1$
    if $b.m = 0$ then
        $HASH\_REMOVE(co.buckets, z)$
    else
        $\bar{b}_{m+1} \leftarrow b.\bar{b}$
        $b.\bar{b} \leftarrow \bar{b}_{m+1} - \frac{1}{b.m}(s(t) - \bar{b}_{m+1})$
        $S_{m+1} \leftarrow b.S$
        $b.S \leftarrow S_{m+1} - (s(t) - \bar{b}_{m+1}) \times (s(t) - b.\bar{b})$
    end
    $co.n \leftarrow co.n - 1;$
    $CalculateCMS(co)$
end
```

Algorithm 7: *CalculateCMS*

```
Input: $co$
begin
    $sum \leftarrow 0$
    $nr\_buckets \leftarrow 0$
    for $b \in co.buckets$ do
        $sum \leftarrow sum + \frac{b.m}{co.n} \times \sqrt{\frac{b.S}{b.m}}$
        $nr\_buckets \leftarrow nr\_buckets + 1$
    end
    $co.cms \leftarrow \frac{1}{nr\_buckets} \times sum$
end
```

6.3 Complexity Analysis

In this section, runtime complexity and space complexity are presented for updating incrementally the $PCC(s,r)$ and $CMS_w(s,r)$.

Lemma 1. *Let s and r be two time series. Incrementally updating $PCC(s,r)$ takes $\mathcal{O}(1)$ time.*

Proof. Algorithm 2, 3 and 4 are executed in a runtime of $\mathcal{O}(1)$ each, since a constant number of basic arithmetic computations are performed. The overall runtime complexity for incrementally updating $PCC(s,r)$ results in $\mathcal{O}(1)$. □

Lemma 2. *Let s and r be two time series. Incrementally updating $CMS_w(s,r)$ takes $\mathcal{O}(\frac{(max(r(t))-min(r(t))}{w})$ time.*

Proof. The search of bucket b_z in the Hash Table in Algorithm 5 and 6 is done in a average runtime $\mathcal{O}(1)$. If there is no bucket with the bucketID b_z, it adds a new bucket to the hash table with a average runtime $\mathcal{O}(1)$. If the bucket is empty after removing the value, the bucket will be deleted in the hash table with a runtime $\mathcal{O}(1)$ and updating $\bar{b}$ and S will be done in $\mathcal{O}(1)$ as well. Calculating the CMS in Algorithm 7 goes then through all buckets in the hash table. Therefore, the runtime depends on the number of buckets used to compute the CMS. The maximum number of buckets can be computed by dividing the difference of the biggest value in the reference time series r and the smallest value in the reference time series r through the bucket width w. Therefore, the overall runtime complexity for $CMS_w(s,r)$ results in $\mathcal{O}(\frac{(max(r(t))-min(r(t))}{w})$ □

Regarding space complexity, assume both time series s and r are loaded into ring buffers in main memory, each of size $\mathcal{O}(L_W)$. Although PCC and CMS are computed incrementally, measurements in the sliding window W have to be kept in the main memory because once the sliding window advances, the measurement which drops out of the sliding windows has to be removed from the PCC and CMS.

Lemma 3. *Let s and r be two time series. The space complexity for incrementally calculating $PCC(s,r)$ is $\mathcal{O}(L_W)$.*

Proof. As shown before, maintaining r and s in two ring buffers has space complexity $\mathcal{O}(L_W)$. The PCC-Object consists of a constant number of elements to compute the PCC. Therefore, the overall space complexity for $PCC(s,r)$ results in $\mathcal{O}(L_W)$. □

Lemma 4. *Let s and r be two time series. The space complexity for incrementally calculating $CMS_w(s, r)$ is $\mathcal{O}(L_W + \frac{(max(r(t))-min(r(t)))}{w})$.*

Proof. Similar to the space complexity for incrementally calculating $PCC(s, r)$, the time series r and s are maintained in two ring buffers with a space complexity $\mathcal{O}(L_W)$. A single bucket consists of a constant number of elements to calculate the standard deviation and has therefore space complexity $\mathcal{O}(1)$. However, space complexity for calculating $CMS_w(s, r)$ depends on the number of buckets used. Therefore, the overall space complexity for $CMS_w(s, r)$ results in $\mathcal{O}(L_W + \frac{(max(r(t))-min(r(t)))}{w})$. □

7 Experimental Evaluation

7.1 Sliding Window

In this Section we conduct experiments to study the impact of the sliding window's size L_W on the ranking of the time series and its runtime. The goal is to find the optimal sliding window size L_W. A value for L_W is optimal if it is the smallest possible value that produces a stable ranking. We look for the smallest possible value L_W, as to retain the least amount of data in main memory. For all experiments, unless otherwise noted, the temperature time series of Montana has been randomly chosen as the base time series s and the temperature time series of Sitten as the reference time series r. The measured unit is degree Celsius.

In Figure 13 we show the development of $PCC(s, r)$ for r over a time range of $L_W = 24$ months. Similarly in Figure 14, we show how $CMS_w(s, r)$ for $w \in \{0.5, 1\}$ changes over the same time frame and time series. In Figure 13 we observe that for $L_W < 3$ months PCC behaves very unstable and assumes values between 0.4 and 0.8. The same pattern is observable for CMS_w in Figure 14 where $CMS_{0.5}$ fluctuates between 0.035 and 0.07 and $CMS_{1.0}$ between 0.075 and 0.0125 respectively. This might be due to outliers weighted stronger for a small sliding window L_W.

An identical pattern occurs for PCC and CMS for a sliding window between three and ten month, i.e. $3 \leqslant L_W < 10$. PCC and CMS are approaching their stable values during this window but behave still unstable. Both, PCC and CMS reach their stable, observable values $PCC = 0.95$, $CMS_{0.5} = 0.025$ and $CMS_{1.0} = 0.06$ for a sliding window greater or equal 15 months, i.e. $15 \leqslant L_W$, as indicated in Figures 13 and 14. These results show that weather data in general has a strong linear relationship over a certain time frame and it has a direct influence on finding the reference time series r which is most similar with the base time series s, i.e. being ranked first (see Figures 15, 16 and 17). Thus we summarize that reliable results should be considered with a set holding data over 15 months or 64'800 measurements.

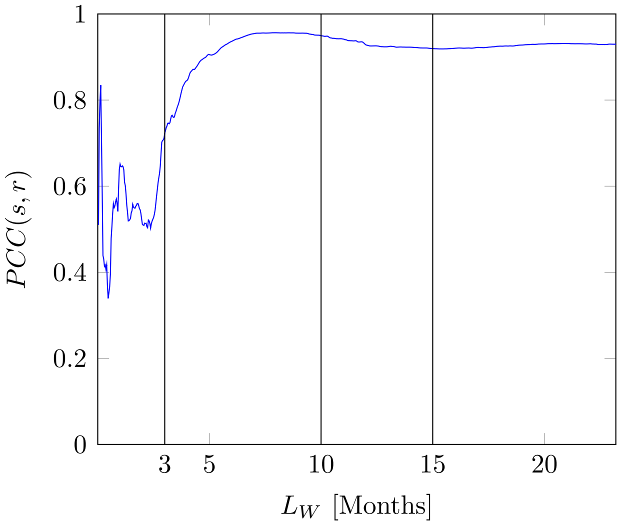

Figure 13: Development of $PCC(s,r)$ with increasing L_W.

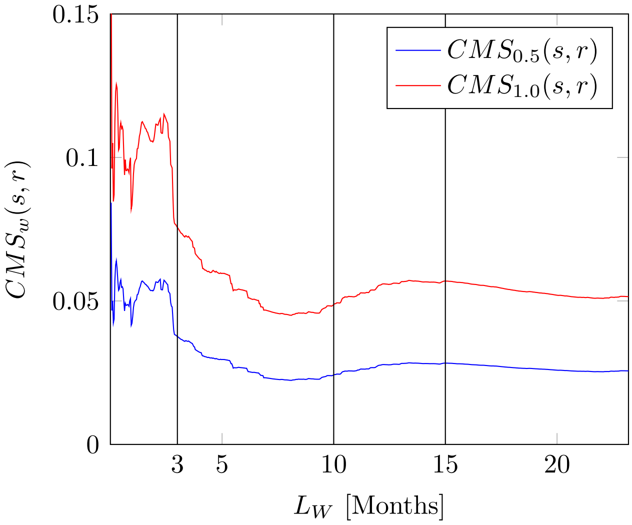

Figure 14: Development of $CMS_w(s,r)$ with increasing L_W.

Figures 15 to 17 show the development of the ranking of nine reference time series as L_W increases. The figures indicate the size of the sliding window L_W on the x-axis and the position of a reference time series in the rankings from one to nine on the y-axis. We can observe that since PCC and CMS fluctuate heavily for $L_W < 3$ months, the rankings behave correspondingly. As discussed before, since PCC and CMS have not stabilized yet for $10 \leqslant L_W < 15$, the rankings in Figures 15 to 17 are still fluctuating. Finally PCC and CMS stabilize for $L_W \geqslant 15$, time series' rankings are not supposed to alter as the sliding window increases over 15 months. This is true for the PCC (see Figure 15). The rankings of CMS in Figures 16 and 17 show that the ranking changes even if $L_W \geqslant 15$ months. This occurs if two reference time series are very similar to each other, e.g. Samedan and Scuol, which both are located in the same area facing the same environment.

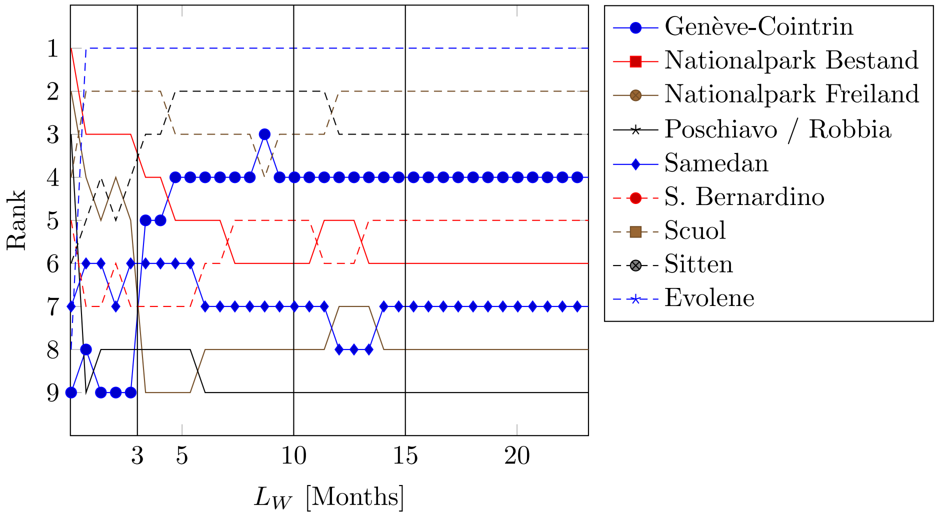

Figure 15: PCC Ranking change with increasing L_W.

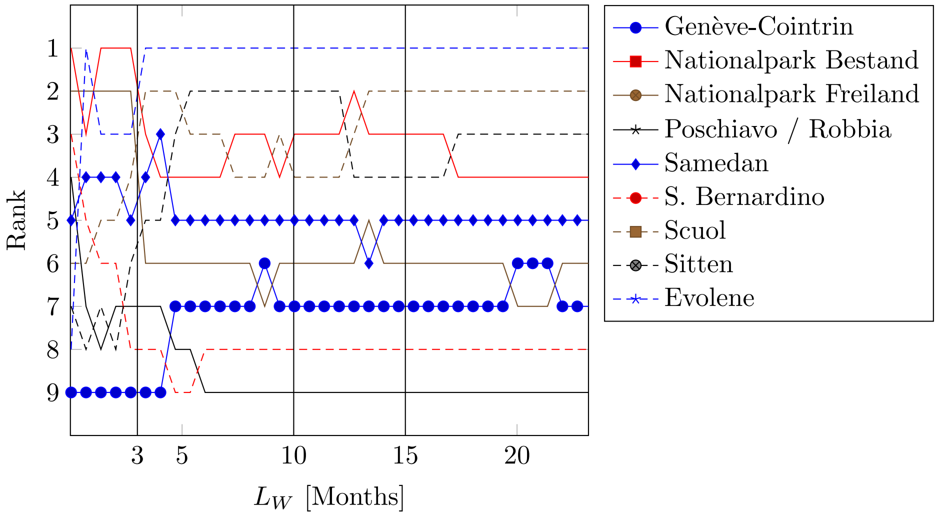

Figure 16: $CMS_{0.5}$ ranking change with increasing L_W.

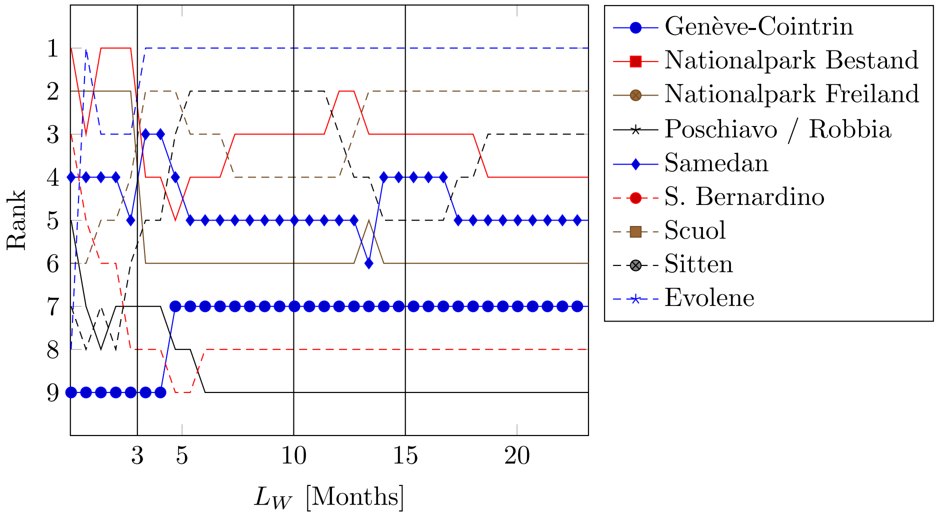

Figure 17: $CMS_{1.0}$ ranking change with increasing L_W.

In Table 8, we can observe that both CMS rankings are almost identical (Figures 16 and 17) . Thus the bucket width w has in this settings very little influence on the rankings. We can further conclude that the rankings of $CMS_{0.5}$ and $CMS_{1.0}$ are identical for $L_W = 24$ months. However this is not true for PCC with $L_W = 24$ months, whose rankings are different from $CMS_{0.5}$ and $CMS_{1.0}$ for $L_W = 24$ months, as Figures 15 to 17 indicate.

Rank	Station	PCC	Station	$CMS_{0.5}$	$CMS_{1.0}$
1	Evolene	0.975	Evolene	0.017	0.034
2	Scuol	0.947	Scuol	0.022	0.044
3	Sitten	0.945	Sitten	0.024	0.048
4	Genève-Cointrin	0.943	Nat.Park Bestand	0.024	0.048
5	S. Bernardino	0.936	Samedan	0.025	0.049
6	Nat.Park Bestand	0.924	Nat.Park Freiland	0.026	0.052
7	Samedan	0.920	Genève-Cointrin	0.027	0.055
8	Nat.Park Freiland	0.919	S. Bernardino	0.030	0.059
9	Poschiavo/Robbia	0.902	Poschiavo/Robbia	0.034	0.067

Table 8: Overview of PCC and CMS rankings over a sliding window $L_W = 24$.

7.2 Weather Phenomena

In this section we introduce three different weather conditions that can be found in our data set: the Föhn, the Bise and Temperature Inversion. Then we execute PCC and Case Matching Similarity (CMS) on these phenomena to see how the similarity changes.

7.2.1 Föhn

Föhn is a common weather phenomena in mountainous regions. Föhn winds are caused by the subsidence of moist air after passing a high mountain. The air is forced to move upslope when it encounters a mountain barrier. As the temperature decreases with height, the moist air will become saturated and condense to form clouds and rain when it rises to a certain height. After passing the mountain barrier and descending along the leeside (downwind side) of the mountain, the air becomes warmer. Temperature of drier air will rise even faster. This results in dry and hot winds. According to the Federal Office of Meteorology and Climatology MeteoSwiss [6] Föhn leads to a rapid increase of the temperature.

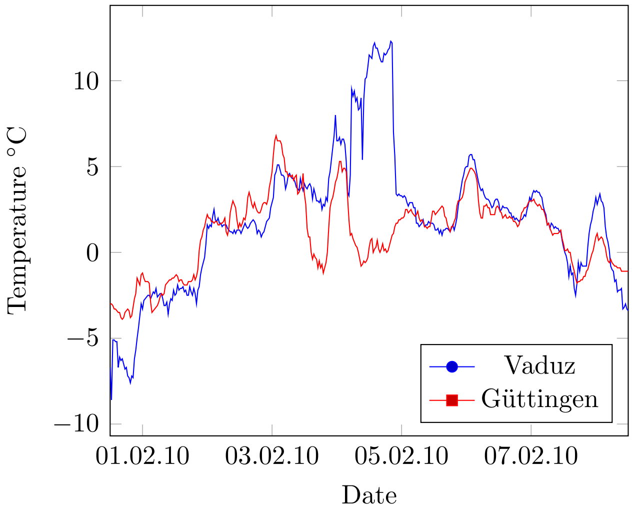

Figure 18: Temperature Güttingen / Vaduz Feb 2010.

Figure 18 shows the temperature curve of Vaduz and Güttingen in a time frame of one week. Vaduz lies in a typical Föhn-Valley whereas Güttingen lies rather in lowlands. As we see in Figure 18, the temperature in both places is quite similar, however the temperature in Vaduz at the 4. February 2010 shoots up in a very short time. There is even no significant temperature decrease during the night. On the other hand, the temperature in Güttingen significantly goes down in the night and goes up again during the day. This rapid temperature increase is a typical characteristics of the weather phenomena Föhn.

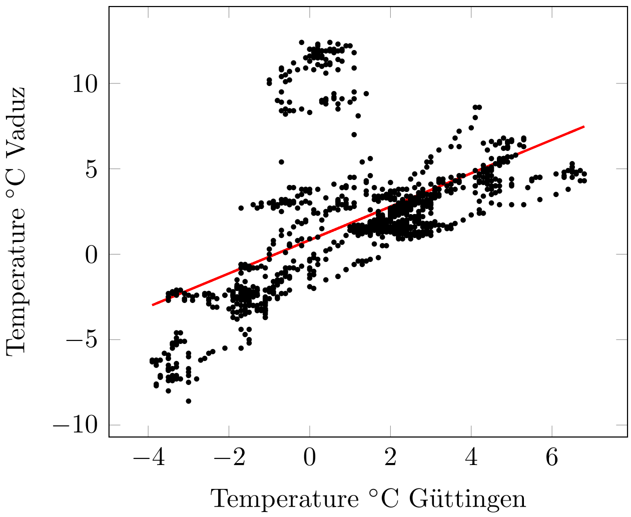

Figure 19: Scatter plot Temperature Güttingen / Vaduz Feb 2010.

In Figure 19 we see the temperature scatter plot from Vaduz and Güttingen between 1.2.2010 and 8.2.2010. A linear relationship between these two locations is evident. However, a cluster of outliers is conspicuous. The reason for this pattern is the weather phenomena Föhn on the fourth and fifth February 2010. The warm winds from the Föhn led the temperature in Vaduz increase noticeably with respect to the temperature in Güttingen.

Table 9 contains the result of calculating the PCC and CMS of the two temperature time series of Vaduz and Güttingen. According to [7], a PCC of 0.55 indicates a moderate linear correlation between Vaduz and Güttingen. Nevertheless, the ranking shows that Vaduz is rather a bad reference time series for Güttingen. Comparing the PCC ranks from the same time period with the previous three years and the following three years, it shows that the ranking of the PCC has only been worse in 2008, when Vaduz was ranked 49th. The other five times it has been ranked 17th (2007), 35th (2009), 18th (2011), 10th (2012) and 8th in 2013. This shows that during a Föhn period, the linear correlation between two time series diminishes and causes

	Result	#Values	Rank
PCC(Vaduz, Güttingen)	0.557172	1152	48/78
$CMS_{0.5}$(Vaduz, Güttingen)	0.088440	1152	55/78
CMS_1(Vaduz, Güttingen)	0.187341	1152	58/78

Table 9: Overview of PCC and CMS_w of Vaduz / Güttingen in Feb 2010.

a noticeable change in the rankings. Considering the CMS ranks, there is no noticeable change in the rankings which is caused due to the Föhn. In 2007 $CMS_{0.5}$ (respectively CMS_1) ranked Vaduz as 45th (46th), in 2008 as 54th (53th), in 2009 as 55th (54th), in 2011 as 27th (31th), in 2012 as 73th (71th) and in 2013 as 77th (73th). Table 10 shows a summary of of the rankings.

	PCC Rank	$CMS_{0.5}$ Rank	CMS_1 Rank
2007	17	45	46
2008	49	54	53
2009	35	55	54
2010	48	55	58
2011	18	27	31
2012	10	73	71
2013	8	77	73

Table 10: PCC and CMS_w ranking of Güttingen / Vaduz in Feb 2007 – 2013.

7.2.2 Bise

Bise is a cold, dry wind from northeast which blows through the Swiss Midland. It is caused by canalization of the air-current along the northern edge of the Alps, during high-pressure conditions in northern or eastern Europe respectively. Towards western Midland, the Bise is pressed between Jura and Pre-Alps whereby it strengthens and mostly climaxes on the western shore of Lake Geneva because the distance between the Alps and Jura mountains gets smaller to the west. In summer, Bise wind causes rather dry and sunny weather whereas in winter, it frequently forms low stratus clouds over the Midland by strengthening the inversion layer [6].

In Figure 20 the barometric pressure of Güttingen and Genève-Cointrin in May 2014 is shown. As the diagram shows, in the western part of Switzerland there is mostly a higher barometric pressure. This results in wind from south west through Midland. However, on 15.5.2014, Güttingen has a slightly higher barometric pressure than Genève-Cointrin – which led to a cold wind from northeast, the so called Bise.

Figure 21 shows the scatter plot of the barometric pressure of Güttingen and Genève-Cointrin in May 2014. The red line in the figure represents the linear regression line. The scatter plot shows a very interesting pattern of the two time series. As we have seen in Figure 20, the Bise occurs, when the barometric pressure in Güttingen lies in the interval $[970, 980]$ hPa. The

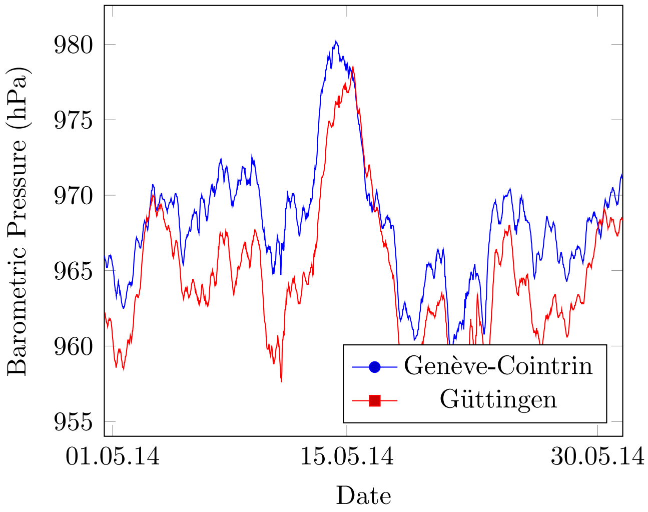

Figure 20: Barometric Pressure Güttingen / Genève-Cointrin May 2014.

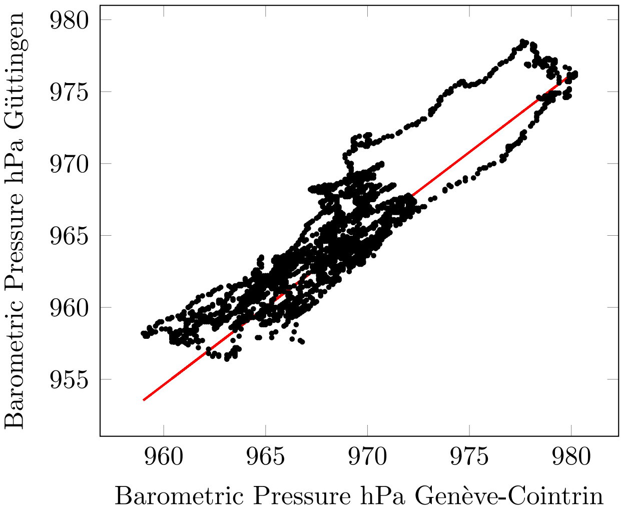

Figure 21: Scatterplot Barometric Pressure Güttingen / Genève-Cointrin May 2014.

pattern in top right corner in Figure 21, when Genève-Cointrin's barometric pressure is between [972, 980], is a result of the increasing barometric pressure in Figure 20. The reason is that Güttingen usually has roughly 5-10 hPa lower barometric pressure than Genève-Cointrin, but during the Bise period, the barometric pressure suddenly increases and reaches or even surpasses that of Genève-Cointrin.

	Result	#Values	Rank
PCC(Genève-Cointrin, Güttingen)	0.907644	4464	1/110
$CMS_{0.5}$(Genève-Cointrin, Güttingen)	0.030262	4464	1/110
CMS_1(Genève-Cointrin, Güttingen)	0.063294	4464	1/110

Table 11: Overview of PCC and CMS_w of Genève-Cointrin / Güttingen in May 2014.

Table 11 contains the calculated PCC and CMS measures. Noteworthy is the very high PCC of 0.907644 which indicates a very high linear correlation of the two time series despite the pattern on the top right. This high correlation is because weather data in general do have a strong linear relationship over a certain time frame, especially barometric pressure time series. Moreover, we found that the rather short periods of time in which this weather phenomena occur are too short to have a high impact on the PCC or CMS, because in the remaining period a strong linear correlation exists.

The PCC and CMS measures of the two time series is less interesting due to the fact that the two time series are similar. From 2011 till 2015 Genève-Cointrin has always been ranked as the top reference time series for Güttingen with some exceptions. PCC has ranked Genève-Cointrin on the second place in 2013. $\text{CMS}_{0.5}$ has ranked Genève-Cointrin on second place in 2012 and 2013 and CMS_1 in 2013. Table 12 shows a summary of of the rankings for May from 2011 –2015.

7.2.3 Temperature Inversion

Temperature inversion is the reversal of the normal behavior of temperature in the troposphere, in which a layer of cool air at the surface is overlain by a layer of warmer air. A common cause for a temperature inversion in Switzerland is high inversion fog which is a result of the Bise. The cold air is blown from the northeast into the Swiss Midland under the lighter, warm air and remain there. The result is an inversion layer. In lower regions, the weather is cold and cloudy. Above the fog, the weather is less cold with clear blue sky [6].

	PCC Rank	$CMS_{0.5}$ Rank	CMS_1 Rank
2011	1	1	1
2012	1	2	1
2013	2	2	2
2014	1	1	1
2015	1	1	1

Table 12: PCC and CMS_w ranking of Güttingen / Genève-Cointrin in May 2011 – 2015.

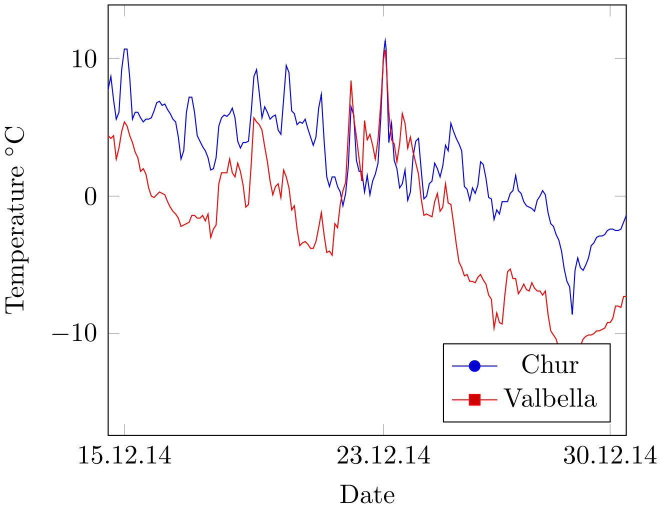

Figure 22: Temperature Chur / Valbella Dec 2014.

The chart in Figure 22 illustrates an example of a temperature inversion. The blue line of represents the temperature of Chur, which is 556 meters above sea level. The temperature of Valbella is represented by the red line. Valbella is 1569 meters above sea level. In mid December 2014, the temperature in Valbella is clearly colder than in Chur. Around the 22. December 2014, the temperature in Valbella starts to increase while the temperature in Chur stays stable. For the next couple of days, a temperature inversion is present. It is significantly warmer in Valbella than in Chur. Beside the warmer temperature in Valbella during the day, also the night is warmer than in Chur. On 22. December 2014, the warmest measured temperature in Valbella was 8.4 °C and in Chur 6.4 °C. During the night, the temperature then decreases in Valbella to 3.2 °C whereas the temperature in Churs goes to 0.1 °C.

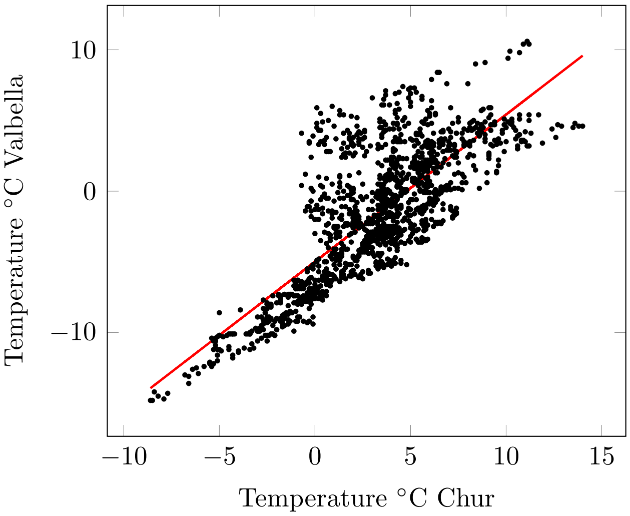

Figure 23: Scatter plot Temperature Chur / Valbella Dec 2014.

The temperature inversion has also an impact on the scatter chart in Figure 23. In the left part of the scatter chart when Chur's temperature is between [-10, 0] degree celsius, the temperature of both locations correlate linearly. On the right side of the scatter chart, no linear correlation is visible, that is, the points scatter in a wide range around the regression line. This is due that the temperature in Valbella is increasing whereas the temperature in Chur stays on the same level.

Table 9 contains the result of calculating the PCC and CMS of the two temperature time series of Chur and Valbella in December 2014.

Temperature inversion is very common weather phenomena in the winter. Considering the same time period from 2011 till 2015, in every time period

	Result	#Values	Rank
PCC(Chur, Valbella)	0.775177	4320	25/110
$CMS_{0.5}$(Chur, Valbella)	0.032744	4320	9/110
CMS_1(Chur, Valbella)	0.067276	4320	10/110

Table 13: Overview of PCC and CMS_w of Chur / Valbella in December 2014.

at least one temperature inversion occurs. However, considering Table 14 the ranking of all three measures are rather volatile from 5th place till 45th place. Therefore, temperature inversion does only have a very small impact on the PCC, respectively on the CMS. This is because the time period during a temperature inversion occurs is rather short and during the remaining period a linear correlation exists.

	PCC Rank	$CMS_{0.5}$ Rank	CMS_1 Rank
2011	16	37	38
2012	7	5	5
2013	26	45	45
2014	25	9	10
2015	31	28	29

Table 14: PCC and CMS_w ranking of Chur / Valbella in December 2011 – 2015.

7.3 Runtime Analysis

As shown in Section 5.3, the bucket width has an impact on the number of buckets used to compute the CMS. In addition, the bucket width has an impact on the number of values which are stored on average within one bucket and as proven in Lemma 2 (Section 6.3), the overall runtime complexity for $CMS_w(s, r)$ results in $\mathcal{O}(\frac{(max(r(t))-min(r(t))}{w})$, which mean in other words that a higher bucket width w leads to a smaller runtime.

To perform the runtime analysis of the CMS, we ran our algorithm ten times and measured the average runtime, excluding the time for fetching the data from the database. For our experiment we use the temperature time series from Montana as a base time series and the temperature time series from Sitten as a reference time series. We compute the runtime over a time frame from 01.01.2006 to 26.09.2009 (1000 days). Using a sliding window length $L_W = 15$ months. In Sitten, the highest measured temperatures was 35.9 °C and the lowest measured temperature was -12.8 °C.

Figure 24 presents the average runtime of ten runs of $CMS_w(s,r)$, as we increase bucket width w from 0.1 °C to 10 °C. $CMS_{0.1}(s,r)$ has the highest average runtime with 986.9 milliseconds.

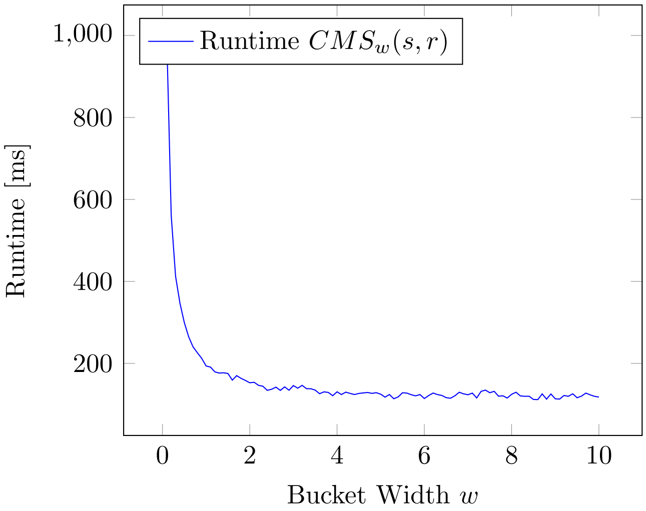

Figure 24: Runtime in milliseconds of CMS$_W$(s,r) with increasing bucket width w.

Doubling w reduces the runtime almost by half, $CMS_{0.2}(s,r)$ has an average runtime of 561.4. From a bucket width w of 4, the runtime is flattening out with a runtime average of 121.7 milliseconds and the runtime remains mostly stable with increasing bucket width w. The lowest average runtime has been measured with a bucket width of 8.5.

The reason for the high runtime with a small bucket width is because a small bucket width leads to more bucket. The algorithm for calculating CMS requires to loop over every bucket. Consequently, a higher number of buckets leads to a higher runtime.

8 Conclusion

Our work introduces into time series analysis and its application on a real data set. The goal is to analyze and compare the similarity (or correlation) of a base time series to a reference time series. Finally the reference time series shall be ranked according to its similarity to the base time series. Therefore we studied and implemented two similarity functions: The *Pearson Correlation Coefficient (PCC)* (Section 4) and the *Case Matching Similarity (CMS)* (Section 5). We introduced first PCC as a *symmetric* similarity measure to automatically rank a reference time series to a base time series. We have formally shown the connection between PCC and linear regression. More specifically, we have shown how PCC can be interpreted as quality measure for the obtained regression line. PCC is able to detect linear correlations only. Its range reaches from maximum +1, which represents a perfect, positive linear relationship to a minimum of -1, representing a perfect, negative linear relationship. If the two variables do not linearly correlate at all, the PCC's value is zero.

CMS in contrary is also able to detect non-linear correlation. CMS splits the range of the reference time series into equal-sized buckets. Each bucket contains the values of s such that the corresponding value of r is within the bucket limits. The CMS range is $[0, \infty)$, with 0 denoting maximum similarity. CMS bucket width allows a small degree of adaption to the nature of the time series at the same time the bucket width has a direct impact on the number of buckets which leads to different results and runtime. We provided rough guidelines how to choose bucket width w.

Section 6 explains how the two algorithms are implemented. Since weather data's time series are by definition infinite and memory capacity is limited, the sliding window keeps only data in memory of a chosen time frame. This allows to implement efficient algorithms and to calculate PCC and CMS incrementally. The algorithms space and memory complexity have been analyzed.

In Section 7 we first analyzed the impact of the sliding window's size on the resulting PCC and CMS. Thereby we found out that both PCC and CMS produce unstable results for small streaming windows. The results are stable applying a sliding window over more than 15 months. Thus reliable results should be considered with a data set holding at least data over 15 months. Further we analyzed in the same section impacts of three weather phenomena on the PCC and CMS rankings. Since all of them occur on rather short time ranges and data should be analyzed at least for a time frame of more than 15 months, their impact on the rankings is negligible small. The last experiment analyzed the impact of the bucket width of CMS on the algorithm's runtime. We found out that a smaller bucket width leads to more buckets. A higher amount of bucket intensifies calculations and increases the algorithm's runtime.

References

[1] K. Wellenzohn, M. H. Böhlen, A. Dignös, J. Gamper, and H. Mitterer, "Continuous imputation of missing values in streams of pattern-determining time series," in *EDBT*, 2017.

[2] S. Chatterjee and A. S. Hadi, *Regression analysis by example.* Hoboken, N.J: Wiley-Interscience, 2006.

[3] K. Wellenzohn, "Imputation of missing values in highly correlated streams of time series data," Master's thesis, Free University of Bolzano, 2015.

[4] F. Smarandache, "Alternatives to pearson's and spearman's correlation coefficients."

[5] T. Finch, "Incremental calculation of weighted mean and variance," tech. rep., University of Cambridge, February 2009.

[6] Bundesamt für Meteorologie und Klimatologie MeteoSchweiz, "Typische Wetterlagen im Alpenraum," 2015.

[7] M. Khayati and M. H. Böhlen, "REBOM: recovery of blocks of missing values in time series," in *Proceedings of the 18th International Conference on Management of Data, COMAD 2012, 2012, Pune, India*, pp. 44–55, 2012.

Appendices

A Solving Least Squares

Rewriting Equation 13 and 14 we obtain

$$\sum_{t\in W} s(t) = L_w\beta_0 + \beta_1 \sum_{t\in W} r(t) \text{ and} \tag{41}$$

$$\sum_{t\in W} s(t)r(t) = \beta_0 \sum_{t\in W} r(t) + \beta_1 \sum_{t\in W} r(t)^2. \tag{42}$$

Dividing both sides of Equation 41 by L_W, we get

$$\bar{s} = \beta_0 + \beta_1\bar{r} \text{ or}$$

$$\beta_0 = \bar{s} - \beta_1\bar{r}.$$

Substituting β_0 in Equation 42, we get

$$\begin{aligned}\sum_{t\in W} s(t)r(t) &= (\bar{s} - \beta_1\bar{r}) \sum_{t\in W} r(t) + \beta_1 \sum_{t\in W} r(t)^2 \\ &= \bar{s} \sum_{t\in W} r(t) - \beta_1\bar{r} \sum_{t\in W} r(t) + \beta_1 \sum_{t\in W} r(t)^2.\end{aligned}$$

Solving for β_1 we obtain:

$$\beta_1 = \frac{\sum_{t\in W} s(t)r(t) - \bar{s}\sum_{t\in W} r(t)}{\sum_{t\in W} r(t)^2 - \bar{r}\sum_{t\in W} r(t)} = \frac{\sum_{t\in W}(s(t) - \bar{s})(r(t) - \bar{r})}{\sum_{t\in W}(r(t) - \bar{r})^2}.$$

B The Sum of Residuals

We want to show that the sum of residuals $\sum_{t\in W} \epsilon_t = 0$. We substitute ϵ_t and obtain

$$\sum_{t\in W}(s(t) - \hat{s}(t)) = 0$$

$$\sum_{t\in W}(s(t) - \beta_0 - \beta_1 r(t)) = 0$$

$$\sum_{t\in W} s(t) = L_w\beta_0 + \beta_1 \sum_{t\in W}(r(t))$$

Notice the last equation corresponds to Equation 41 in Appendix A, where we have shown how to choose β_0 and β_1 with the least squares method to satisfy this equation.

C Coefficient of Determination

We want to show that $2\sum_{t\in W}\epsilon_t(\hat{s}(t)-\bar{s})=0$, which can be rewritten as

$$\sum_{t\in W}\epsilon_t\hat{s}(t)-\bar{s}\sum_{t\in W}\epsilon_t=0$$

The second term, $\hat{s}\sum_{t\in W}=0$ as we have shown in Appendix B. Therefore we need to show that

$$\sum_{t\in W}\epsilon_t\hat{s}(t)=0.$$

We substitute $\hat{s}(t)$ and obtain:

$$\sum_{t\in W}\epsilon_t\hat{s}(t)=\sum_{t\in W}\epsilon_t(\beta_0+\beta_1 r(t))=\beta_0\sum_{t\in W}\epsilon_t+\beta_1\sum_{t\in W}\epsilon_t r(t)=0.$$

Again, from Appendix B we know that $\beta_0\sum_{t\in W}\epsilon_t=0$ and therefore we need to show that $\beta_1\sum_{t\in W}\epsilon_t r(t)=0$. We substitute ϵ_t and obtain

$$\beta_1\sum_{t\in W}(s(t)-\hat{s}(t))r(t)=\beta_1\sum_{t\in W}(s(t)-\beta_0-\beta_1 r(t))r(t)=0.$$

Notice that this last equation is equivalent to Equation 14 and that due to the least squares method (*see Appendix A*), values β_0 and β_1 were chosen such that this equation is satisfied.