Decision Analytics with Heatmap Visualization for Multi-step Ensemble Data

An Application of Uncertainty Modeling to Historical Consistent Neural Network and Other Forecasts

With today's computing power, it is easy to generate huge amounts of data. The real challenge lies in adequately condensing the data in decision making processes. Here, the focus is on ensemble data that typically arises when distributions of forecasts are generated for several time steps in the future. Often a distribution is aggregated by taking an ensemble's mean or median. This results in a single line that is easy to interpret. However, this single line may be seriously misleading when the ensemble splits into two or more different bundles. The mean or median may also lie in a region where there are only very few ensemble members. To remedy this, a heatmap visualization to better represent ensemble data for decision analytics is proposed. Heatmap visualization provides an intuitive way to identify regions of high and low activity. The regions are color-coded according to the (weighted) number of ensemble members in a specific region.

DOI 10.1007/s12599-014-0326-4

The Authors

Dipl.-Math. Cornelius Köpp
Prof. Dr. Hans-Jörg von Mettenheim (✉)
Prof. Dr. Michael H. Breitner
Institute for Business Information Systems
Leibniz University Hannover
Königsworther Platz 1
30167 Hannover
Germany
mettenheim@iwi.uni-hannover.de

Received: 2013-07-15
Accepted: 2014-01-31
Accepted after two revisions by the editors of the special focus.
Published online: 2014-04-26

This article is also available in German in print and via http://www.wirtschaftsinformatik.de: Köpp C, von Mettenheim H-J, Breitner MH (2014) Decision Analytics mit Heatmap-Visualisierung von mehrschrittigen Ensembledaten. Eine Anwendung von Unsicherheitsmodellierung für Historical Consistent Neural Network und andere Prognosetechniken. WIRTSCHAFTSINFORMATIK. doi: 10.1007/s11576-014-0417-3.

Electronic Supplementary Material
The online version of this article (doi: 10.1007/s12599-014-0326-4) contains supplementary material, which is available to authorized users.

1 Introduction and Motivation

One of the typical tasks of a decision support system is to produce forecasts and – more importantly – to help people interpret those forecasts. However, single value forecasts may be misleading. For this reason ensemble forecasts – based on a collection of several individual forecasts – can be used to largely improve the forecasting accuracy; see for example (Zhang and Berardi 2001). With ensemble forecasts, quite often only a single value is (or a few points of the distribution are) used. This is useful for automatic usage in information systems but the real shape of the other features of the distribution remains unclear; see for example (Welch 2001; Hansen 2008). It is not necessarily the case that the forecast distribution is unimodal. Generally it could be multimodal. That means that we cannot characterize the forecast by a single number (for example the mean). Several different models make up the forecast and we cannot expect our forecast to have a single peak. In this case the mean or median can be misleading, as we will show. Multi-step means that our forecast does not just forecast, for example, tomorrow's value. Rather, the forecasts we are looking at will stretch over several time-steps. In essence, common data aggregation techniques either lose information or do not scale well. In both cases this greatly reduces the usefulness of the forecast.

We present first steps towards answering the research question: *"How can an adequate visualization enable decision analytics for today's ensemble forecast methods?"*

Ensembles of artificial neural network (ANN) models are a typical case where we obtain forecasts that consist of several hundred individual paths. In the present paper we look at a 20-day forecast for the price of natural gas in US dollars. The figures are computed from an ensemble of 200 networks. We are using a new class of ANN, the Historical Consistent Neural Network (HCNN) intro-

duced by Zimmermann et al. (2010). See also von Mettenheim and Breitner (2010) for a detailed presentation and performance evaluation. For the goals of the present paper it is sufficient to be aware of the fact that HCNNs use a simple state equation to compute the following state from the immediately preceding state. Multi-step forecasts are therefore easy to generate. We generally use HCNNs when we have to model several time-series and their distribution simultaneously. When we train different HCNNs with randomly initialized weights, we obtain a diverse ensemble of forecasts.

The exact forecast asset is not central to the following discussion and it is not our goal to evaluate the forecast performance of the ANN. Rather, our focus is on making the forecast output easier to interpret for the human decision maker. Our approach is therefore not limited to ANNs. Other model types that might produce an unlimited number of forecast paths based on good historical performance could also be used in this context. These include, for example, Support Vector Machines, Evolutionary Programming, and Monte Carlo Simulations. For this reason we outline the ANN model only briefly. In our paper we present a heatmap visualization of the resulting ensemble forecasts. This is a step towards visually supporting the human decision maker, because heatmaps aggregate information but conserve the essence of the forecast, even if the distribution is multimodal.

We can now make our research question more concrete: "*How can we intuitively present the complete forecast information to a decision maker, but also exploit all distribution information?*"

We propose heatmap visualization. A heatmap allows us to differentiate between more active and less active regions of the forecast space by color coding (see **Figs. 1(d)–1(f)** and examples in **Figs. 3**, **4**, and **5**). A detailed discussion of **Fig. 1** will follow below, especially in Sect. 6.

Whereas simply plotting aggregate values in **Figs. 1(a)** and **1(b)** loses information, it is also not possible to only plot each forecast individually as is done in **Fig. 1(c)**. We cannot distinguish individual forecasts anymore and the output is useless. **Figures 1(d)** and **1(e)** show our heatmap approach and **Fig. 1(f)** is an even more useful presentation. We can clearly see splitting paths in **Figs. 1(d)–1(f)**. This is a warning signal from the model: the forecast is dubious.

Heatmaps also help us gauge the quality of the forecast. Depending on the width of the forecast and the number of peaks we see in the heatmap, we can qualify the forecast as more or less reliable This offers an alternative to the usual binary output of most of today's forecasting methods. It is quite common that a forecast model either outputs "up" or "down". This is not entirely honest. There should be a third output possibility: "don't know". Heatmaps offer just that: rather than hiding information behind a single number (which will invariably be wrong) they present the entire forecast spectrum to the decision maker. Problematic areas are then easy to identify.

2 Research Design

To answer the research question, we were inspired by the Design Science Research approach from Hevner et al. (2004, p. 83). **Figure 2** shows the implementation of our research design. In the following we place our contribution in the context of selected research guidelines:

The presented research is relevant (*Problem Relevance*), as our visualization approach tackles distributional forecasts that for example arise in the context of ANN ensembles. The same visualization could be useful for any other forecast models that generate ensembles, or time-series. We review existing concepts in the area of ensemble and time-series visual representation and show the limitations of commonly used visualization techniques in Sect. 3. In Sect. 4 we present the formal model of our method for heatmap generation. We argue that this approach is an artifact that "extends the boundaries of human problem solving" (Hevner et al. 2004) because it can help decision makers arrive at better decisions by providing additional information, which is invisible when using statistic aggregations (*Design as an Artifact*).

The first step of our evaluation is to build a prototype implementation. The prototype is described in Sect. 5. We use the prototype to demonstrate the utility of our approach, in the form of qualitative information gain. One example scenario is given by the above-mentioned gas-price forecast scenario; another is artificially created for clearer description of the information gained by interpolation,following the concept of Descriptive *Design Evaluation* with Scenarios (Hevner et al. 2004, p. 86).

An earlier stage of this work was presented to the scientific and practice audience at a conference (von Mettenheim et al. 2012) and a workshop (*Communication of Research*). We discussed the results with ANN experts and incorporated their feedback. From this discussion an independent implementation of our approach emerged, conducted by expert users in a large international company. Hevner et al. (2004) states that "the objective of design-science research is to develop technology-based solutions to important and relevant business problems". We consider the existence of an independent implementation to be a strong indicator for the importance and business *relevance* of the problem we address with our approach.

3 Related Work

Potter et al. (2009b) underline the "enormous power" of ensemble data sets, but also the "formidable challenge" of ensemble visualization due to their complexity. Andrienko and Andrienko (2005) focus on spatially distributed time-series data as used in cartographic and geo-visualization applications. They criticize the combined plotting of many localized time-series, as cluttering and overlapping lines result in a hardly legible display, and the concept becomes "completely unusable" for a large number of hundreds and more time-series. As an alternative they propose map- and aggregation-based visualization. Aggregation is realized by plotting the minimum, maximum, median and quartiles. Andrienko et al. (2010) focus on event detection support in multiple time-series. They present a toolkit with interactive user-controlled data visualization, using mean or median for statistical summary of multiple time-series. Bade et al. (2004) use minima, maxima, median and the 25 % and 75 % percentile for the presentation of aggregated high-frequency data streams. Hao et al. (2009) describe a visual support framework for time-series prediction. Their tool uses a one dimensional heatmap style "visual accuracy indicator" to show over, under, and close predictions.

Aigner et al. (2007) present a "conceptual visual analytics framework for time-oriented data". They describe aspects

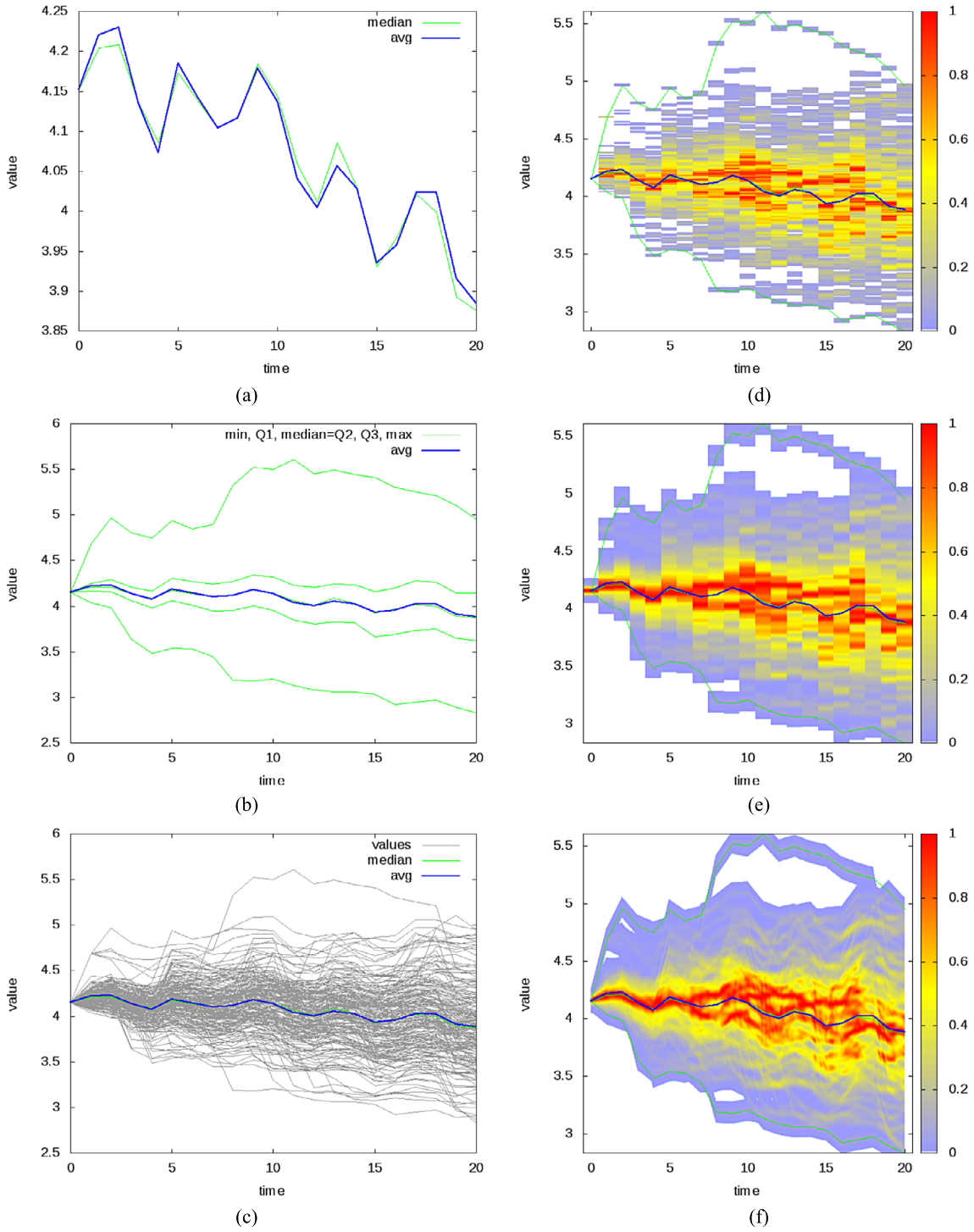

Fig. 1 Visualizations of a multi-step ensemble forecast

of time visualization and time-oriented data, and illustrate some of them. The exploring of trends and patterns is referred to as "particularly important tasks when dealing with time-oriented data and information". They see interaction as an essential part of visualization. May et al. (2010) also emphasize the definition of visual analytics and refer to it as "the science of analytical reasoning facilitated by interactive visual interfaces." Thomas and Cook (2006) follow the same direction. Savikhin et al. (2011) presents a visual analytics tool for financial decision support.

Buono et al. (2007) present a similarity-based time-series forecasting approach on a data set of historical time-series. Their interactive time-series visualization tool "Time Searcher 3" uses a "River Plot View" and presents statistical attributes and quantiles for the indication of forecast uncertainty. Feng et al. (2010) present several methods for visualizing uncertainty, motivated by medical application. They characterize density plots as "useful tools for summarizing extremely

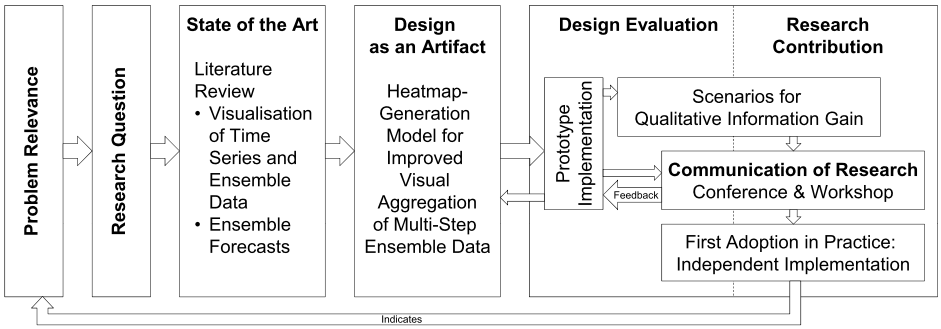

Fig. 2 Research design inspired by Hevner et al. (2004, p. 83)

large data sets" and propose them as the "fundamental tool for visualizing uncertain multivariate data". Their visualization technique for scatter plots is based on kernel density estimation (KDE) of probability density in a data set.

Yagi et al. (2012) presents a data visualization technique for plotting tagged time-varying data in the single polyline chart space. To deal with reduced readability of "hundreds or even thousands" of lines, they use a two-step approach: they cluster the polylines and select representative samples for each cluster. They implement an interactive visualization environment.

In summary, the visualization of time-series ensembles by plotting summary data, especially in the simple form of mean/median/extrema/quantiles, is commonly used; see for example (Andrienko and Andrienko 2005; Andrienko et al. 2010; Buono et al. 2007; Feng et al. 2010). The most basic approach is to use the mean or median (**Figs. 1(a)**). Adding some representative percentiles such as quartiles (including extrema) increases the visible information (**Fig. 1(b)**). An additional step is to plot every path of individual ensemble members; see also Potter et al. (2009a). **Figure 1(c)** is, for example, a plot of 200 individual paths. The approach of plotting every ensemble member is inefficient When the ensemble includes several hundred members it becomes unwieldy. The "readability of jammed lines is a common problem of information visualization techniques, and several works have addressed the problem" (Uchida and Itoh 2009).

We additionally searched for similar approaches in Google Scholar and Google Images (for visual impression) using combinations of at least two of the following keywords: ensemble; {visualisation|visualization}; uncertainty; heatmap; time-series; distribution.

The concept of fanplots/fancharts is similar to our approach, as a heatmap-like representation is used for uncertainty visualization. Fanplots are used in macroeconomic or probabilistic population forecasts, for example. The United Kingdom Monetary Policy Committee (MPC) presents the projections of GDP and CPI inflation as fancharts in their inflation reports. Elder et al. (2005) document this usage in detail: The charts are based on the mode ("the single most likely point") as a central projection and a symmetrical or skewed distribution. The colored area is defined to cover 90 % of the future outcome. The width of the fan chart defines the degree of uncertainty and widens with the forecast horizon, "reflecting the increased probability that some unforeseen event could" occur (Elder et al. 2005, p. 330). Raymer et al. (2012) use fanplots for the visualization of probabilistic population forecasting. The second author published the package *fanplot* for the R Project of Statistical Computing. The plot is generated from the calculated "percentiles for a set of sequential distributions over a specified time period" (Abel 2013), so there is also the possibility to generate fanplots from ensemble data. An introduction and three exemplary use cases are presented by Abel (2012). However, the fanplot concept is limited, because it is based on percentiles and the mode: The information of multimodal distributions and therewith the possible forecast result "don't" is lost.

4 Mathematical Model of Heatmap Generation

Table 1 defines the formal base of the heatmap generation approach. In the following we will use the term forecast data, although the model could be applied to all ordered multi-step ensemble data. Therefore the process steps do not need to describe time.

Two parameters define the dimension of forecast data F: m gives the number of ensemble members used to generate the distribution. The parameter n gives the number of forecast steps. M and N are the related index sets that are needed to describe the position in the forecast data, where N is extended by the index 0 for the already known starting point. The value of the starting point is given by g. The input data for heatmap generation D is composed of F by adding the step 0 and set the value to g for all ensemble members.

The parameter h directly influences horizontal resolution of the resulting heatmap. For $h > 1$ the forecast data is expanded by the linear interpolation of each ensemble member $D^{\cdot h}$. $N^{\cdot h}$ gives the associated index set of interpolated forecast steps. This interpolation preserves the original forecast data and we have $N^{\cdot 1} = N$ and $D^{\cdot 1} = D$.

The heatmap resolution is defined by the number of interpolated steps $n{\cdot}h$ and the vertical resolution parameter y_{res}. X and Y are the related index sets of heatmap coordinates, V is the set of values associated to the y-coordinates. The value range is defined as $[v_{\min}, v_{\max}]$ and should contain at least all values of D. When generating heatmaps from a series of forecasts, the same range should be used for all heatmaps and should contain all values of all forecasts for better comparability.

The heatmap H is generated column by column for every time-step and interpolated time-step. Each column H_x of the heatmap is independent of all other columns and the calculation of the heatmap column generator function c depends on data column $D^{\cdot h}_{x/h}$ only. Consequently, the interpolation does not alter the non-interpolated columns. Basically, c can be any function of the type

Table 1 Formal symbols and definitions

Symbol and definitions	Description
$m \in \mathbb{N}$	Number of ensemble members
$n \in \mathbb{N}$	Number of forecast steps
$F \in \mathbb{R}^{m \times n}$	Ensemble forecast
$M := \{1; \ldots; m\} \subset \mathbb{N}$	Index set of ensemble members
$N := \{0; \ldots; n\} \subset \mathbb{N}_0$	Index set of forecast steps including starting point
$g \in \mathbb{R}$	Value of starting point (before forecast)
$D := (gF) = (D_0 \ldots D_n) = (d_{ij}) \in \mathbb{R}^{M \times N}$	Input data (columns) for heatmap generation
$h \in \mathbb{N}$	Forecast interpolation factor ($h = 1$ no interpolation)
$N^{\cdot h} := \{0; \frac{1}{h}; \ldots; 1; \ldots; n - \frac{1}{h}; n\} \subset \mathbb{Q}_+$	Index set of interpolated forecast steps
$D^{\cdot h} := (d_{ij}), i \in M, j \in N^{*h}$ with $d_{i,j'+q} := (1-q) \cdot d_{i,j'} + q \cdot d_{i,j'+1}$ $(j'+q) \in N^{\cdot h}, j' \in N$	Interpolated data
$x \in X := \{0, \ldots, n \cdot h\} \subset \mathbb{N}_0$	Heatmap x-coordinates
$y \in Y := \{0, \ldots, r\} \subset \mathbb{N}_0$	Heatmap y-coordinates
$v_{\min}, v_{\max} \in \mathbb{R}$ with $d_{ij} \in [v_{\min}, v_{\max}]$	Range of forecasted values shown in heatmap
$V := \{v_y := v_{\min} + y \cdot (v_{\max} - v_{\min})/r \mid y \in Y\} \subset \mathbb{R}$	Heatmap y-coordinate values
$H := (H_0 \ldots H_{n \cdot h}) \in \mathbb{R}^{Y \times X}$ with $H_x := c(D^{\cdot h}_{x/h}) \in \mathbb{R}^Y$	Heatmap data and heatmap column generator
$c(D^{\cdot h}_j) := (c_j(v_0), \ldots, c_j(v_r))^t \in \mathbb{R}^Y$	Heatmap column generator
$c_j(v) := s_j(v) * k(v) = \sum_{i \in M} k(v - d_{ij}) \in \mathbb{R}$	Heatmap element generation by convolution
$s_j(v) := \lvert\{i \in M \mid d_{ij} = v\}\rvert, j \in N^{\cdot h}$	Count of ensemble members at v in column j
$k(\delta) : \mathbb{R} \to \mathbb{R}$	Convolution kernel
$H_{\text{normed}} := (H_0 \ldots H_{n \cdot h}) \in \mathbb{R}^{Y \times X}$ with $H_{\text{normed},x} := H_x / \max(H_x)$	Heatmap data normed by column

$c : \mathbb{R}^M \to \mathbb{R}^Y$ that maps the distribution of the values in a data column $D^{\cdot h}_{x/h}$ to a "comprehensible" image. The column vector $c(D^{\cdot h}_j)$ is composed by the calculation results of c_j for each heatmap y-coordinate value. For c_j we use a function class that can be reproduced by a (discrete) convolution of kernel $k(\delta)$ with $s_j(v)$ the number of ensemble members at v in column j. We will discuss several realized heat map column generators in the following sections.

The normalization of heatmap columns is the optional last step of heatmap calculation. This could improve the readability, as the starting point at index 0 always has the maximum density. This density is usually much higher than the density of forecasts. The use of normalization adds supplementary requirements to the heatmap column generator: it must be ensured that each column vector H_x consists of non-negative values and includes at least one element with a positive non-zero value. This ensures that $\max(H_x) > 0$ for all $x \in X$, so that $H_{\text{normed},x}$ the division of each heatmap column by its maximum is defined.

The values of the resulting heatmap matrix H (or H_{normed} in the same way) are presented color-coded in the range $[0, \max(H)]$.

5 Prototype Visualization Tool

We implement the heatmap visualization model in a software prototype in Java with SWT (Standard Widget Toolkit) for the user interface. **Figure 3** shows the prototype with an alternative heatmap style and disabled column norming. Overall the current version of the prototype includes 15 different heatmap (column) generators, include algorithms based on counting members within a given ε-environment (simple and very fast), convolutions with gauss and similar shaped kernels, kernel density estimations, plus two differing concepts similar to a cumulative distribution function and its mean gradient in the associated value range of one heatmap pixel. The member counting within a given ε-environment can also be described as a convolution with $k_1(\delta) := 1_{\{|\delta| < \varepsilon\}}$.

The right section of the program window controls several plot parameters. The heatmap style and optional normalization, the vertical plot resolution r ("heatmap v res") and the horizontal interpolation factor h ("heatmap h res*") can all be configured. Higher parameters produce smoother heatmaps, but will increase the time for heatmap calculation. Other controls allow users to change the free parameter used in several algorithms, and the "class radius divider" for setting the parameter ε in $k_1(\delta)$ by the division of $(\max(H_x)$–$\min(H_x))$ for relative, or $(v_{\max}$–$v_{\min})$ for the absolute radius definition.

Several checkboxes are used to select or deselect plotted information: extrema, quantiles, percentiles, all ensemble values with optional highlighting of all members which touched the extrema at least once, the target value for the retrospective analysis of forecast quality and "ensemble behavior", and finally the heatmap.

We especially focus on the interactive component of our prototype, as emphasized by several authors (Aigner et al. 2007; Andrienko et al. 2010; May et al. 2010; Potter et al. 2009b). The first plotting engine with the free software Gnuplot (version 4.2 patchlevel 6) proved too slow: high resolution plots sometimes took more than 30 seconds. Even a direct transfer of binary-coded heat map data via pipe to the input of the Gnuplot process does not lead to a sufficient acceleration. The additional new internal plotting engine reduces this time to less than 0.2 seconds, an increase in plotting speed by two orders of magnitude, and is now being used as the default. The heatmap generation engine

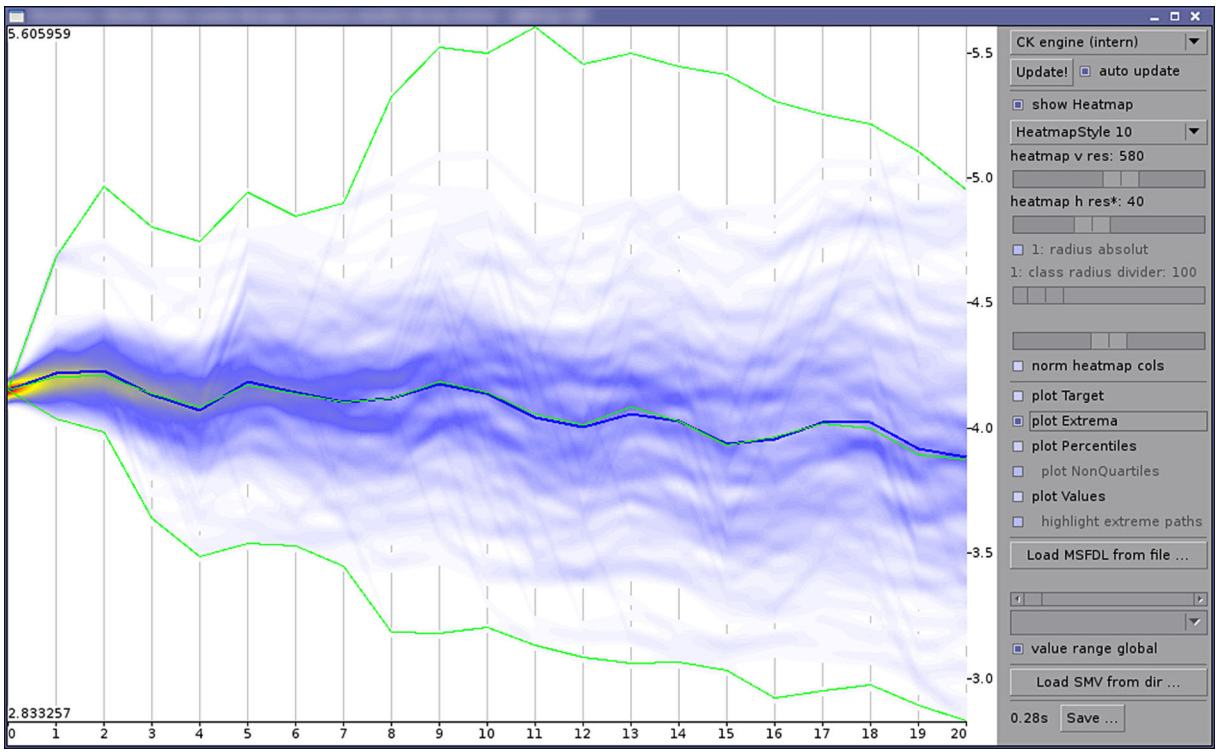

Fig. 3 Screenshot of the prototype visualization tool (alternative heatmap style)

parallelizes computation of the heat-mal columns to the available number of cores and therefore makes good use of multi-core CPUs. The calculation of heatmap columns is simplified, for example by using the gauss kernel $k_6(\delta) := e^{p \cdot \delta \cdot \delta}$ instead of $e^{-\delta^2/(2a^2)}$. An alternative algorithm use a lookup table (pre-calculated only once for all columns) to increase the calculation speed by one order of magnitude. The increase in speed was attained on a common quad-core CPU. We consider the smooth scrolling through large forecast result series as an important aspect. It allows users to see how the forecast develops with time. Also, a fast display of results helps in the context of live applications. In daytrading applications the latency is crucial, for example.

6 Discussion

As a first step of ensemble visualization, we can simply plot the mean or median (see **Fig. 1(a)**). The general visual impression of **Fig. 1(a)** is that of a strong downtrend. The figure does not convey any distributional information. This changes slightly when we add representative percentiles, such as quartiles (see **Fig. 1(b)**). Note an interesting feature: future uncertainty, as measured by the difference of maximum and minimum, does not necessarily increase. In fact, uncertainty decreases during the last five forecast days. The figure now only conveys the visual impression of a slight downtrend, due to the width of the distribution. Nevertheless, we still have no idea of the distribution of individual paths.

An additional step is to plot all the 200 paths of individual ensemble members; see also Potter et al. (2009a). This leads to **Fig. 1(c)**. For clarity, the figure also shows mean and median in bold lines. It becomes apparent that we do not gain much by plotting every path. On the contrary, the information becomes less clear because the paths overlap. All we can see is that the distribution is dense in the middle and less dense at its borders. We might therefore visually conclude that the distribution is unimodal – a possibly dangerous conclusion as we will see later on.

In the heatmap visualization, red regions indicate a forecast of high activity For clarity, **Figs. 1(d)–1(f)** and **Fig. 3** also show mean, median, maximum and minimum in thick lines. **Figure 1(e)** shows a more balanced heatmap style than **Figs. 1(d)**, and **1(f)** shows the heatmap style of **Fig. 1(e)** smoothed by interpolation. We note that often, but not always, the mean coincides with red regions. However, we also note in **Figs. 1(d)** and **1(f)** that approximately from day 5 to day 15 the red region splits into several paths and increases considerably. This becomes especially clear in the smoothed version of the heatmap **Fig. 1(f)**. The conclusion is that according to the forecast the mean is not an accurate representation of the distribution, because (especially for days 6 to 8) the distribution is bimodal or multimodal. This significantly changes the interpretation of the decision maker. Looking only at **Figs. 1(a)–1(c)**, it seems apparent that the forecast for days 6 to 8 is a slight downtrend. Looking at the heatmap we see that the correct answer actually is: the ensemble doesn't know! This is a warning to the decision maker. On the other hand, during the last few forecast days (days 16–20), uncertainty decreases and the model clearly forecasts a downtrend. Keep in mind that we are not dealing with forecast accuracy, we are just exploiting the forecast information a priori.

At first glance no additional information seems to be gained by interpolating between different forecast steps. However, the smoothed graphics allows us to follow the paths in the forecasts (see **Figs. 4** and **5** for the information gained by interpolation). **Figure 4** shows the improved visibility of a splitting path in

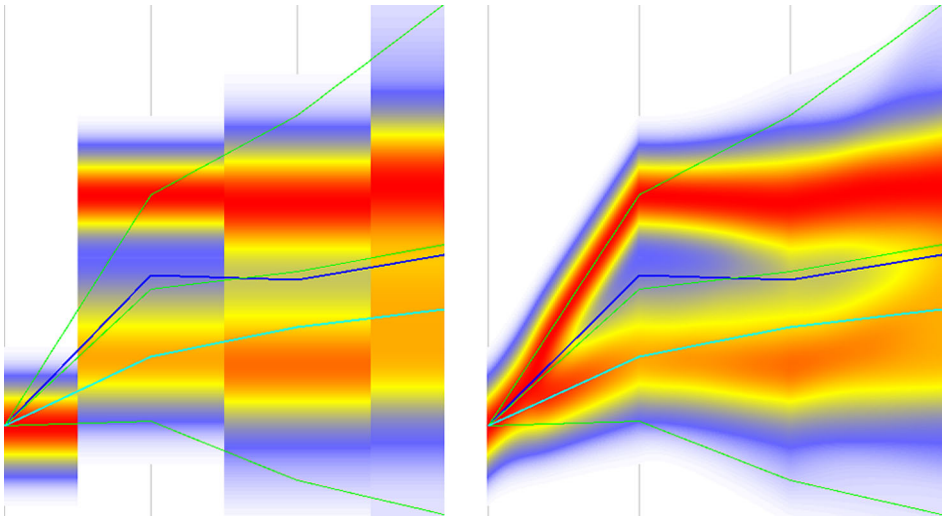

Fig. 4 Information gained by time-step interpolation: splitting paths (without/ with interpolation)

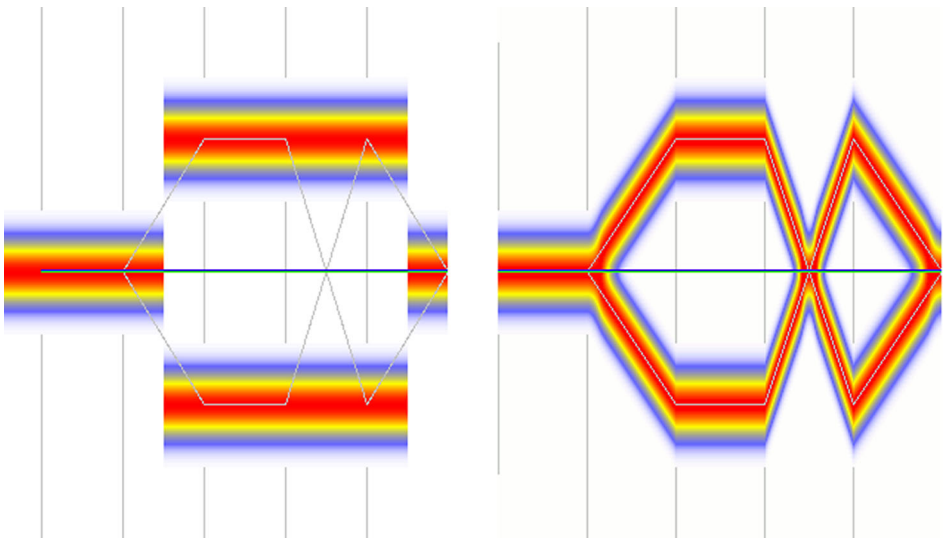

Fig. 5 Information gained by time-step interpolation: crossing paths (without/ with interpolation)

the forecast ensemble, and **Fig. 5** shows an artificial ensemble of two parallel and crossing members. A distinction is possible only with interpolation (see **Figs. 1(e)** without and **Fig. 1(f)** with interpolation).

We implemented 15 heatmap generation algorithms. The numbering reflects the order of implementation. Some of them provide similar visual results; others create a completely different visual impression. Table 1-A in the Appendix (available via http://link.springer.com) shows a formal overview of all implemented styles. Figure 1-A in the Appendix shows the visual result of our example generated with default configuration. Changing the parameters could significantly change the result. In general, that is desired for optimizing the visibility of distribution, but not for all parameters. While indicator-function-based kernels (like k_1 and k_4) are very fast, they have the disadvantage of not producing a stable result under variation of the vertical heatmap resolution r. Continuous (but not necessarily differentiable) kernel functions are more robust and avoid this problem.

Overall, the gauss kernel k_6 generates satisfying results for several data sets, but the calculation of this kernel is slow. To improve the interactivity, we implemented three similar shaped kernels k_7, k_8 and k_9. The kernel k_7 generates the most similar results and is about four times faster; the others are slower and generate a suboptimal result. Finally the pre-calculation of gauss kernels and use of a lookup-table (k_{10}) as well as the reduction from double to float accuracy (k_{11}) could reduce the calculation time by about factor ten without a visible influence. The gauss kernels with lookup-table are independent of the heatmap resolution, provide a good visibility of paths and internal structures at the same time and are fast enough for an interactive use.

The four implemented KDE kernels (k_{11}, k_{12}, k_{13}, k_{14}) have no configurable parameters. This can be an advantage for ease of use, but in the data sets we tested they lost the structure of ensemble distribution as they are "too wide". Kernel k_3 differs from other heat map generators as the result corresponds to a cumulative distribution function. This makes it an interesting choice when the under- or overrun of quantiles is important and enables visualizations such as fanplots.

An apparent alternative to heatmap visualization would be a 3D plot. This would replace or extend the color-coded information with height information. However, the disadvantage of three dimensional plots is that parts of the plot may often hide other parts (see **Fig. 6**). Generally, obtaining adequate information from a three-dimensional plot is

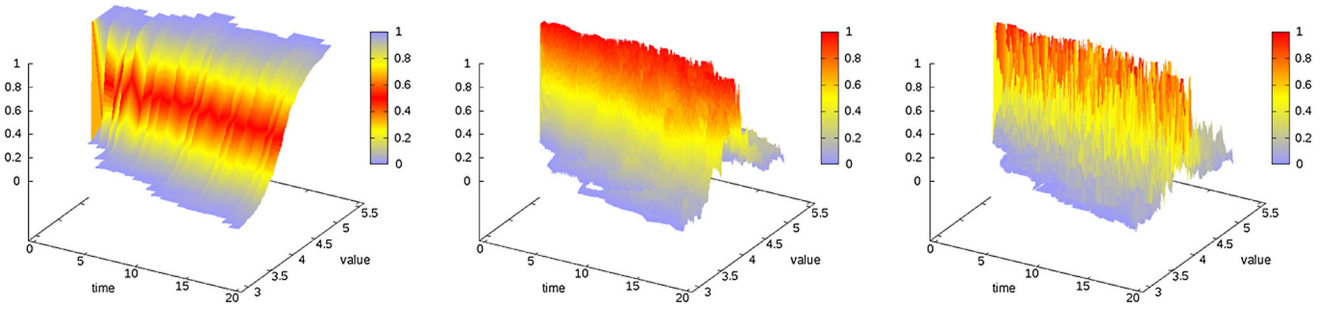

Fig. 6 3D visualization for comparison: the results are more difficult to interpret

more difficult. The 2D heatmap visualization provides a good compromise between information density and visual interpretability.

The underlying forecast model (an ANN example) is not central to the applicability of our research. Our proposed heatmap could be used to visualize all kinds of forecast and in general all ordered data if about 100 or more paths exist. The only assumption is that the forecast is composed of a distribution that cannot be characterized by just a few simple numbers. Nevertheless, heatmaps can also ease the adoption of ANN based forecast methods.

7 Limitations

While heatmap visualization compelling advances, certain limitations are inherent to the approach. The most important drawback is that visual results are difficult to quantify. A human decision maker can easily interpret splitting paths as the state of "don't know". However, the question remains as to how to aggregate the visual information into data that is machine readable, but that does not lose the advantage of heatmap visualization.

The paper only focuses on visual representation. It would be interesting to analyze whether we could also quantify the advantage of using the forecast information more efficiently. This involves, e.g., identifying peaks of the distribution and benchmarking a tri-state model forecast (increases, decreases, don't know) against the realization or a forecast based on mean and median only. Other means of better quantifying the forecast information can be devised. Finally, we suggest that some form of clustering algorithm could be used to filter the information. Yagi et al. (2012) describes clustering for the polyline-plot of time-varying data. Ideally, the algorithm would highlight strong paths and discard regions with noise.

A second category of drawbacks involves the number of meta-parameters that can be set to visualize ensemble data. Depending on parameters such as kernel, resolution, and normalizing of columns, quite different visual output can be achieved. Not every parameter combination leads to meaningful results that really improve the interpretability of data. While other (expert) users did not complain about the complexity and were rapidly able to achieve helpful visual results, this kind of software is not just "plug and play". A basic understanding of the effects of different meta-parameters, especially the working of the smoothing kernels, is necessary. Presently, an automatic tuning of meta-parameters is missing.

Column normalization especially is a double-edged sword. On the one hand, normalization improves the visual impression of distributions that often grow "wider" as time advances. As we move to the "right" of the image, the ensemble often becomes larger, which is a measure of uncertainty. Fewer and fewer ensemble members can be found in a specific radius. A representation without normalization will give a noisy impression (see for example **Fig. 3**). But normalization will destroy the information that fewer ensemble members contribute to a visible path. A human decision maker could falsely come to the conclusion that a path is supported by many ensemble members when it is actually only supported by a slight majority. All in all, the normalization feature of the prototype should be carefully used. A horizontal indicator, similar to the one used by Hao et al. (2009, p. 1), or an additional saturation coding in the heatmap could help reduce the risk of misinterpretation.

A potential improvement of human decisions with heatmap visualization has been discussed with experts, but has not yet been validated. The feedback of researchers and practitioners during conference and workshop presentations and in real-world applications is positive. However, a proper empirical validation of the (perceived) usefulness would involve more formal methods, for example based on expert and user questionnaires. Even if a positive effect is assumed by human decision makers, this does not necessarily imply better decisions. It may be difficult to assess the contribution of better data visualization in generally complex business decision processes; see also Potter et al. (2012, p. 239) who describe the evaluation of uncertainty visualization as "double problematic". However, in the special field of short- and medium-term trading, better decisions translate into better returns. Giving traders access to heatmap visualization of asset forecasts could help in assessing the usefulness of the method. Even then it remains difficult to differentiate between the quality of the forecast method itself and the added value of better visualization. In this vein, a purely quantitative validation without human decision makers could be useful. This would necessitate the above-mentioned clustering algorithm to highlight important paths.

8 Conclusions and Outlook

Our paper outlines steps towards adequate visualization and interpretation of ensemble forecasts and distributed ordered data. Heatmaps offer an improved approach to exploit forecast data than current methods. We see that mean and median do not necessarily confer the right information, for example because the distribution may split. This makes mean and median bad representatives for the ensemble values. Forecast uncertainty measured by the width of the ensemble

does not uniformly increase with future time-steps. A split distribution may actually become unimodal again. Heatmap visualization provides a third alternative for the human decision maker: to the outcomes "up" or "down" we add a state of model uncertainty, "don't know". Thus this approach to forecast is more honest, because it reduces the risk of misinterpreting a forecast consisting of a single number. Here, we explicitly acknowledge uncertainty in the forecast and do not rely on just one number.

The prototype offers different heatmap styles which we have only briefly discussed here. We need further experimentation in selecting an appropriate heatmap style and parameters, as parameterization is "vital for ensuring effective visualization" (Aigner et al. 2007). Relatedly, our approach adds interpolation between time steps as a means of obtaining an even better and smoother impression of the underlying paths.

We do not show the realized path of the forecast time-series here because we are focusing on introducing heatmap visualization. In further work it is important to actually also benchmark the heatmap visualization's performance: we have to answer the question as to whether the additional information indeed helps us arrive at better decisions. However, benchmarking is tightly coupled to an underlying forecast model, in this case ANN. Benchmarking would have shifted the focus of the paper towards the forecast qualities of ANN and HCNN.

In further research we plan to provide a corresponding case study in which we apply the results of using heatmap based decision analytics and support for example in an investment decision context. An interesting area of research is also to analyze the impact of heatmap visualizations with a technology acceptance or information system success model. An independent implementation of a German research and consulting group shows that access to the informational content of the ensemble distribution may be promising.

References

Abel GJ (2012) The fanplot package for R. http://gjabel.wordpress.com/2012/08/13/the-fanplot-package-for-r/. Accessed 2013-10-21

Abel GJ (2013) Fanplot: visualisation of sequential probability distributions using fan charts. http://cran.r-project.org/web/packages/fanplot/index.html. Accessed 2013-10-21

Aigner W, Bertone A, Miksch S, Tominski C, Schumann H (2007) Towards a conceptual framework for visual analytics of time and time-oriented data. In: Henderson SG, Biller B, Hsieh MH, Shortle J, Tew JD, Barton RR (eds) Proc 2007 winter simulation conference, Washington, pp 721–729

Andrienko G, Andrienko N (2005) Visual exploration of the spatial distribution of temporal behaviors. In: Proc ninth international conference on information visualisation. IEEE Computer Society, London, pp 799–806

Andrienko G, Andrienko N, Mladenov M, Mock M, Poelitz C (2010) Extracting events from spatial time series. In: Proc 14th international conference on information visualisation. IEEE Computer Society, London, pp 48–53

Bade R, Schlechtweg S, Miksch S (2004) Connecting time-oriented data and information to a coherent interactive visualization. In: Proc ACM conference on human factors in computing systems (CHI'04). ACM, New York, pp 105–112

Buono P, Plaisant C, Simeone A, Aris A, Shneiderman B, Shmueli G, Jank W (2007) Similarity-based forecasting with simultaneous previews: a river plot interface for time series forecasting. In: Proc 11th international conference information visualization. IEEE Computer Society, Zurich. doi:10.1109/IV.2007.101

Elder R, Kaperanios G, Taylor T, Yates T (2005) Assessing the MPC's fan charts. Bank of England quarterly bulletin: autumn 2005. Bank of England, London, pp 326–348

Feng D, Kwock L, Lee Y, Taylor RM II (2010) Matching visual saliency to confidence in plots of uncertain data. IEEE Transactions on Visualization and Computer Graphics 16(6):980–989

Hansen BE (2008) Least-squares forecast averaging. Journal of Econometrics 146(2):342–350. doi:10.1016/j.jeconom.2008.08.022

Hao MC, Janetzko H, Sharma RK, Dayal U, Keim DA, Castellanos M (2009) Poster: visual prediction of time series. In: Proc IEEE symposium on visual analytics science and technology (VAST 2009), pp 229–230

Hevner AR, March ST, Park J, Ram S (2004) Design science in information systems research. MIS Quarterly 28(1):75–105

May R, Hanrahan P, Keim DA, Shneiderman B, Card S (2010) The state of visual analytics: views on what visual analytics is and where it is going. In: Proc IEEE symposium on visual analytics science and technology (VAST), Salt Lake City, pp 257–259

Potter K, Wilson A, Bremer PT, Williams D, Doutriaux C, Pascucci V, Johnson CR (2009a) Ensemble-vis: a framework for the statistical visualization of ensemble data. In: Proc international conference on data mining workshops, pp 233–240

Potter K, Wilson A, Bremer PT, Williams D, Pascucci V, Johnson C (2009b) A flexible approach for the statistical visualization of ensemble data. In: Proc IEEE ICDM workshop on knowledge discovery from climate data, Miami,

Potter K, Rosen P, Johnson CR (2012) From quantification to visualization: a taxonomy of uncertainty visualization approaches. In: Dienstfrey AM, Boisvert RF (eds) Uncertainty quantification in scientific computing. IFIP advances in information and communication technology, vol 377, pp 226–249. doi:10.1007/978-3-642-32677-6_15

Raymer J, Abel GJ, Rogers A (2012) Does specification matter? Experiments with sim-

Abstract

Cornelius Köpp, Hans-Jörg von Mettenheim, Michael H. Breitner

Decision Analytics with Heatmap Visualization for Multi-step Ensemble Data

An Application of Uncertainty Modeling to Historical Consistent Neural Network and Other Forecasts

Today's forecasting techniques, which are integrated into several information systems, often use ensembles that represent different scenarios. Aggregating these forecasts is a challenging task: when using the mean or median (common practice), important information is lost, especially if the underlying distribution at every step is multimodal. To avoid this, the authors present a heatmap visualization approach. It is easy to visually distinguish regions of high activity (high probability of realization) from regions of low activity. This form of visualization allows to identify splitting paths in the forecast ensemble and adds a "third alternative" to the decision space. Most forecast systems only offer "up" or "down": the presented heatmap visualization additionally introduces "don't know". Looking at the heatmap, regions can be identified in which the underlying forecast model cannot predict the outcome. The authors present a software prototype with interactive visualization to support decision makers and discuss the information gained by its use. The prototype has already been presented to and discussed with researchers and practitioners.

Keywords: Decision analytics, Forecast, Visualization, Fuzzy decision making, Uncertainty modeling, Historical consistent neural network (HCNN), Heatmap

ple multiregional probabilistic population projections. Environment and Planning A 44(11):2664–2686. doi:10.1068/a4533
Savikhin A, Lam HC, Fisher B, Ebert DS (2011) An experimental study of financial portfolio selection with visual analytics for decision support. In: Proc 44th Hawaii international conference on system sciences (HICSS). doi:10.1109/HICSS.2011.54
Thomas JJ, Cook KA (2006) A visual analytics agenda. Computer Graphics and Applications, IEEE 46(1):10–13
Uchida Y, Itoh T (2009) A visualization and level-of-detail control technique for large scale time series data. In: Proc 13th international conference information visualisation, Barcelona, pp 80–85. doi:10.1109/IV.2009.33
von Mettenheim HJ, Breitner MH (2010) Robust decision support systems with matrix forecasts and shared layer perceptrons for finance and other applications. In: Proc ICIS 2010, St. Louis, Paper 83
von Mettenheim HJ, Köpp C, Breitner MH (2012) Visualizing forecasts of neural network ensembles. In: Klatte D, Lüthi HJ, Schmedders K (eds) Operations research proceedings 2011, Zurich, pp 573–578. doi:10.1007/978-3-642-29210-1_91
Welch I (2001) The equity premium consensus forecast revisited. Cowles Foundation discussion paper No 1325, University of California, Los Angeles (UCLA), National Bureau of Economic Research (NBER)
Yagi S, Uchida Y, Itoh T (2012) A polyline-based visualization technique for tagged time-varying data. In: Proc 16th international conference on information visualisation, Montpellier, pp 106–111. doi:10.1109/IV.2012.28
Zhang GP, Berardi VL (2001) Time series forecasting with neural network ensembles: an application for exchange rate prediction. The Journal of the Operational Research Society 52(6):652–664
Zimmermann HG, Grothmann R, Tietz C, von Jouanne-Diedrich H (2010) Market modeling, forecasting and risk analysis with historical consistent neural networks. In: Selected papers of the annual international conference of the German Operations Research Society, Munich, pp 531–536

Decision Analytics with Heatmap Visualization for Multi-Step Ensemble Data – An Application of Uncertainty Modeling to Historical Consistent Neural Network and other Forecasts

Cornelius Köpp, Hans-Jörg von Mettenheim, Michael H. Breitner

Business & Information System Engineering (2014) 6(3)

Anhang (verfügbar online über http://springerlink.com)

Appendix A

Table A-1 Heatmap styles

Kernel / style	**Description**
$k_1(\delta):=1_{\{\lvert\delta\rvert<\varepsilon\}}$	"InClassRadiusHeatmapGenerator"
$k_2(\delta)$ has no trivial representation, this style is not based on convolution	"MeanGradientHeatmapGenerator" This style use the mean gradient (within the range associated to each heatmap element) of a function similar to a linear interpolated cumulative distribution function.
$k_3(\delta):=1_{\{\delta>0\}}$	"DistributionFunctionLikeHeatmapGenerator"
$k_4(\delta):=k_1(\delta)$ with $\varepsilon=(v_{max}-v_{min})/r/2$	"InBucketHeatmapGenerator"
$k_5(\delta):=(\delta_5+1)^{-2}$ with $\delta_5:=(\max(\{\min\{\lvert\delta\rvert-(v_{max}-v_{min})/r/2;1\});0\})+1)^{-2}$	"InverseSquareBucketDistanceHeatmapGenerator"
$k_6(\delta):=e^{p\cdot\delta^2}$ with $p<0$	Gauss Kernel
$k_7(\delta):=1/(1+p\cdot\delta^2)$	
$k_8(\delta):=e^{p\cdot\lvert\delta\rvert}$	Replaced δ^2 by $\lvert\delta\rvert$ in Gauss Kernel
$k_9(\delta):=1-\lvert\tanh(\delta)\rvert$	
$k_{10}(\delta):\approx k_6(\delta)$ with lookup table	Gauss Kernel with lookup table
$k_{11}(\delta):\approx k_{10}(\delta)$ with float not double precision	Gauss Kernel with lookup table and float precison
$k_{12}(\delta):=(2\cdot\pi)^{-1/2}\cdot e^{-\delta^2/2}$	KDE Gauss
$k_{13}(\delta):=1/\pi\cdot 1/(1-\delta^2)$	KDE Cauchy
$k_{14}(\delta):=\frac{1}{2}\cdot e^{-\lvert\delta\rvert}$	KDE Picard
$k_{15}(\delta):=\frac{3}{4}\cdot\max(\{1-\delta^2;0\})$	KDE Epanechikov

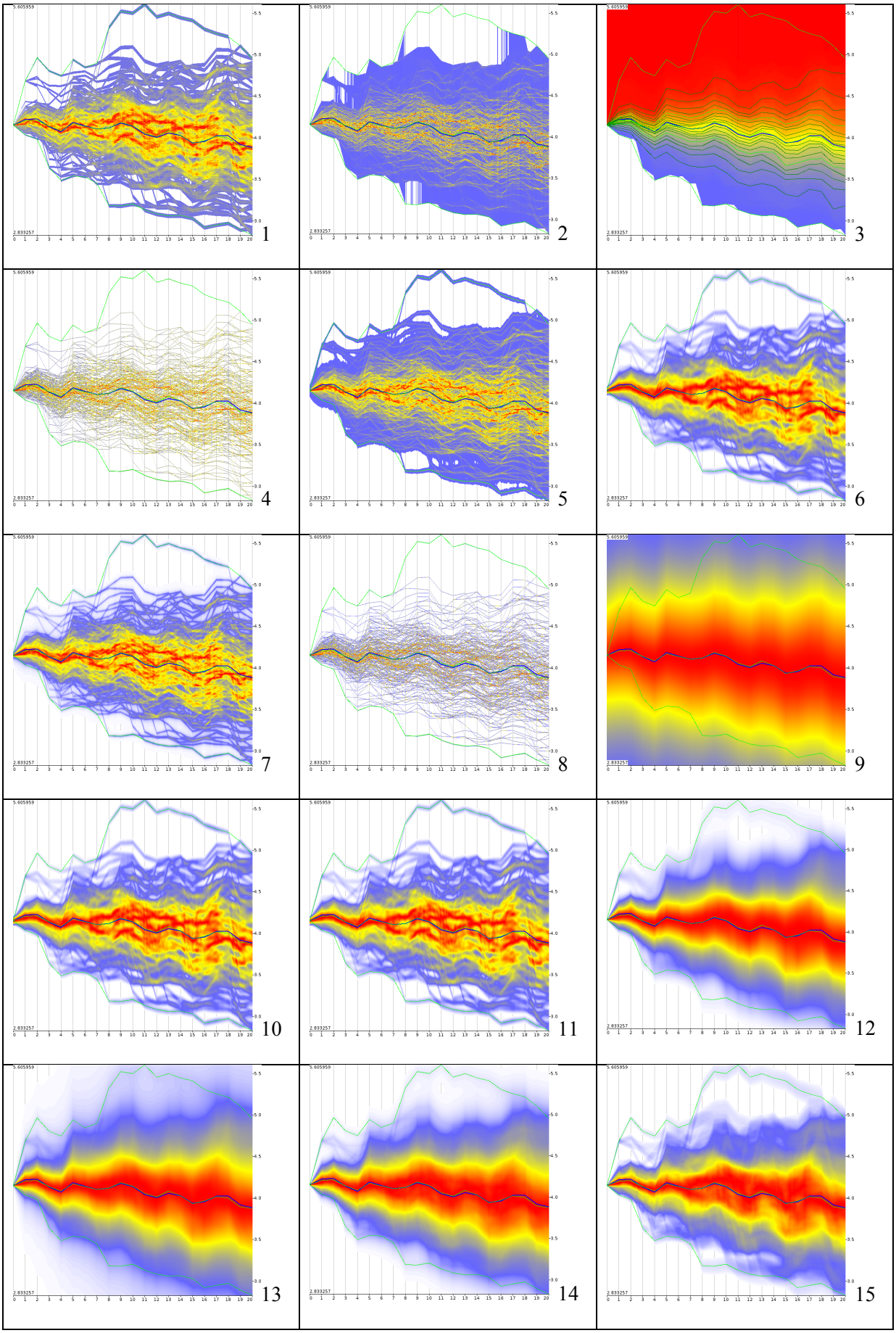

Fig. A-1 Implemented heatmap styles (default parameters, normed)