Lessons Learned in the Challenge: Making Predictions and Scoring Them

Jukka Kohonen — **Univ. of Helsinki**

Jukka Suomela — **Univ. of Helsinki**

PASCAL Challenges Workshop
Southampton, 11 April 2005

Contents:

Part 1: Making Predictions

- *Evaluating Predictive Uncertainty Challenge*
- 5 tasks:
  - 2 classification tasks
  - 3 regression tasks
- *Probabilistic predictions required*

Here we will focus on regression tasks. Our rankings:

- 'Outaouais' *1st*
- 'Gaze' *2nd*
- 'Stereopsis' *last*

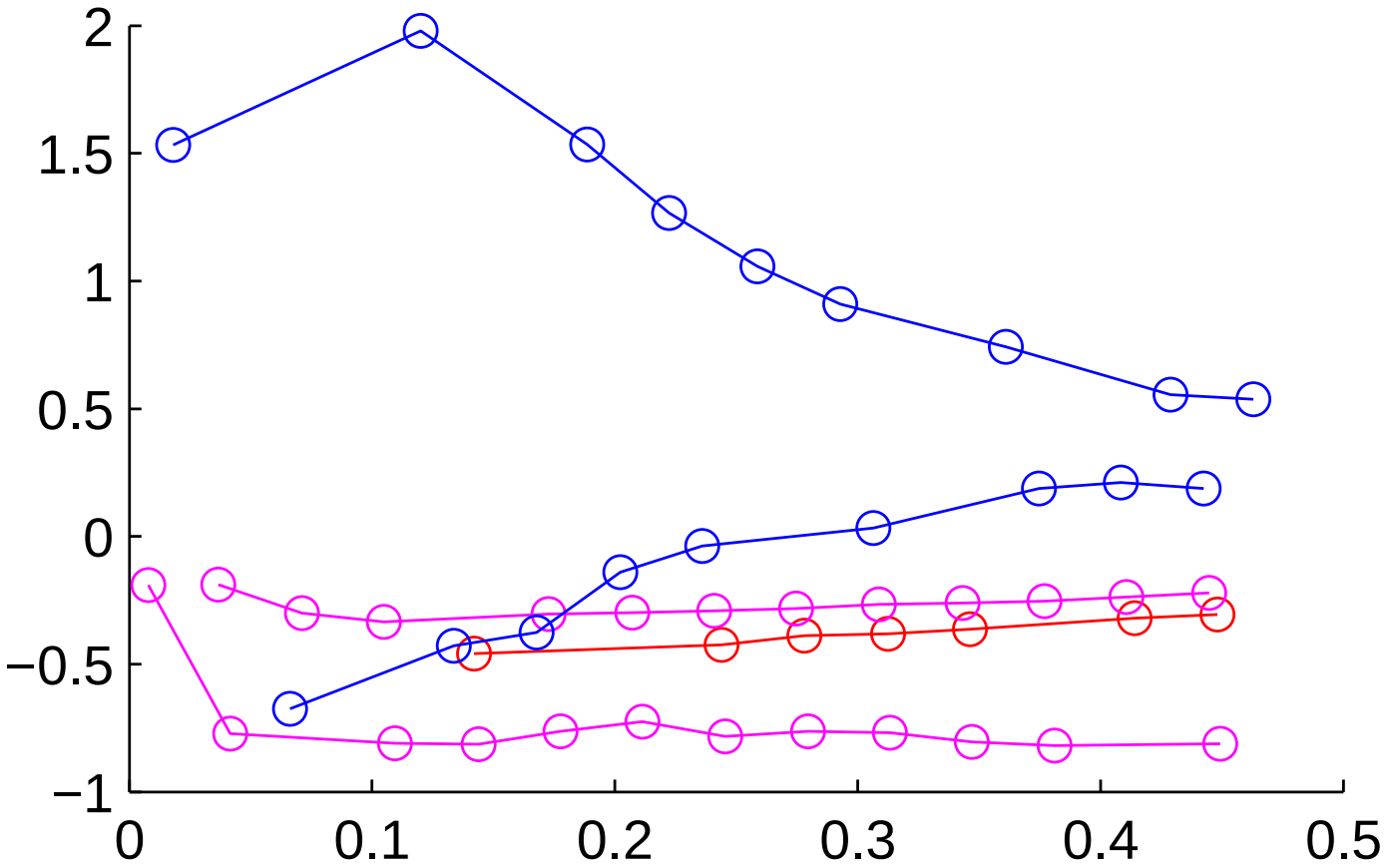
'Outaouais' — Analysis

- Many input variables (37), *very* many training samples (29 000).
- Some input variables were discrete, some were continuous.
- Tried k-nearest-neighbour methods with different values of k, different distance metrics, etc. Very small values of k produced relatively good predictions, while larger neighbourhoods did much worse.
- There seemed to be a surprisingly large number of discrete input variables which were often *equal* for a pair of nearest neighbours.

‘Outaouais’ — Classification

- We ran a piece of software which tried to form a collection of input dimensions which could be used to group all data points into classes.
- Surprising results: We found a set of 23 dimensions which classified all input into about 3 500 classes, each typically containing 13 or 14 points. Almost all classes contained both training and test points!
- For each class, the data points looked as if they were time series data. There was one dimension which we identified as “time”. The target values *typically* changed slowly with time *within each class*.

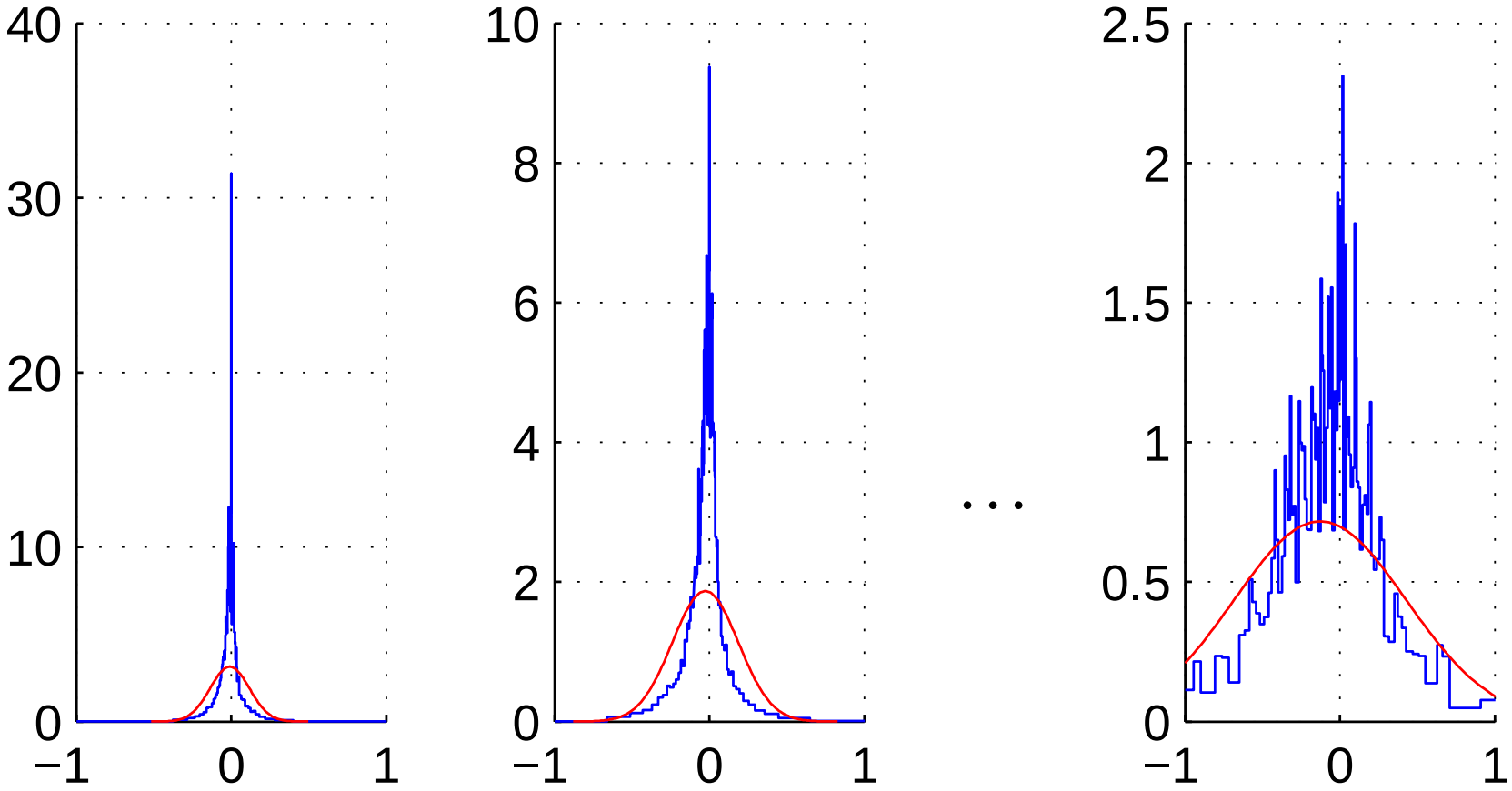
‘Outaouais’ — Classification



'Outaouais' — Statistics

- We could have just fitted a smooth curve within each class. However, in this challenge we needed probabilistic predictions.
- We had 29 000 training points. We were able to calculate *empirical* error distributions for *pairs of samples within one class*, *conditioned* on the discretised distance in the "time" dimension.
- I.e., we first answered this question:
  - If we know that two points, x_1 and x_2, are in the same class, and that the "time" passed between measuring x_1 and x_2 is roughly T, what is the expected distribution of the *difference* of target values y_1 and y_2?

‘Outaouais’ — Statistics



‘Outaouais’ — Predictions

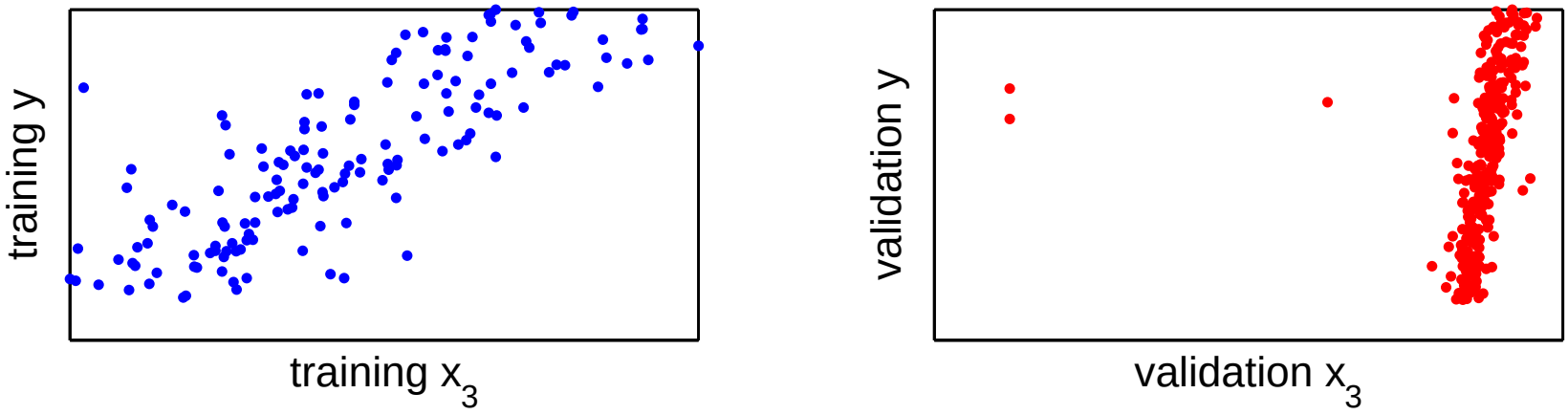
- We calculated 27 (actually 14 + mirror images) empirical distributions, one for each discretised time interval.
- Prediction: Pick 1-NN value within the same class. Calculate distance in the “time” dimension. Discretise distance. Get the corresponding pre-calculated error distributions. Predict the target value of the neighbour *plus* the error distribution.
- This way we got highly non-Gaussian predictions, kurtosis $6 \ldots 22$. We submitted the results as a set of quantiles.

'Outaouais' — Results

- Our mean square error (0.056) was higher than what some other competitors had achieved (0.038). However, the NLPD loss was the lowest (-0.88 for us, -0.65 for the second place). Thus, our predictive distributions were more accurate.
- What can we learn? At least one thing: Surprisingly naive methods may work *if* you can use large amounts of real data to estimate probability distributions.
- Did we model the phenomenon or abuse the data set?

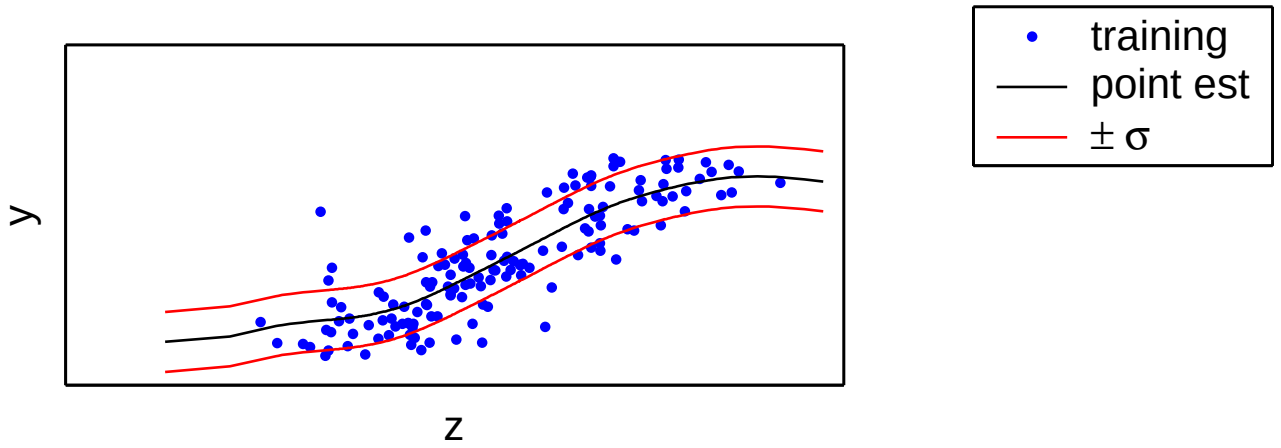
'Gaze' — Input Data

- Input data is 12-dimensional and contains 450 training samples.
- Visual inspection of (x_i, y) for input dimensions $i = 1, \ldots, 12$ reveals clear dependence of y on x_1 and x_3. Other dimensions seem less useful $\implies$ throw them away.
- Some x_3 outliers in validation $\implies$ replace with sample mean.



'Gaze' — Local Linear Regression

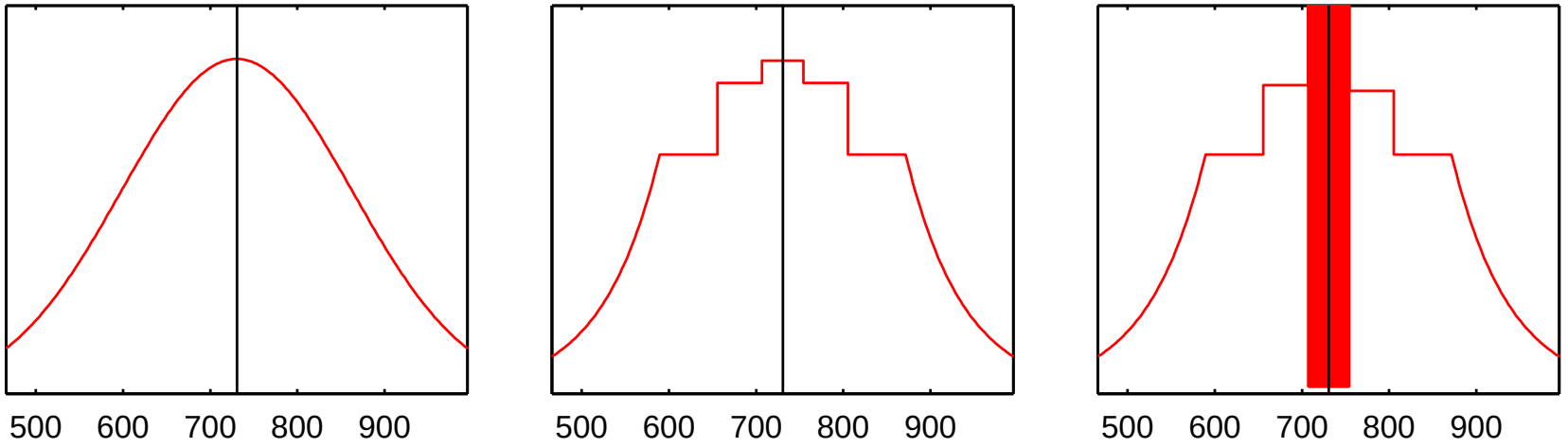
- One-dimensional LLR on linearly transformed input $z = w_1 x_1 + w_3 x_3$, where w chosen by cross-validation.
- LLR gives point estimates; for probabilistic prediction, we assume Gaussian error.



- Error variance estimate = average square residual in training data; tried local and global averaging, global was good enough.

‘Gaze’ — Shaping the Predictions

- Initial idea: $N(\mu, \sigma^2)$ where μ is the point prediction from LLR.
- But targets are integers in the range of $122 \ldots 1000$.
- Discretise the Gaussian into 6 quantiles.
- Replace highest bracket by narrow peaks on integers (peak width $= 2 \cdot 10^{-13}$, density $= 1.5 \cdot 10^{10}$).



‘Stereopsis’ — Input Data

- The data set only had 4 input dimensions.
- Visual inspection of the data showed a clear, regular structure and the name of the data set was an additional hint: “stereopsis” means stereoscopic vision.
- Based on studies, we formed a hypothesis of the physical phenomenon used to create the data.

'Stereopsis' — Model

Assumed model:

- The input data consists of two coordinate pairs, (x_1, y_1) and (x_2, y_2). Each pair corresponds to the location of the image of a calibration target, as seen by a video camera.
- The training targets correspond to the distance z between the calibration target and a fixed surface.
- The calibration target is moved in a $10 \times 10 \times 10$ grid. The grid is almost parallel to the surface from which distances are measured.

No idea if this model is correct, but having some visual model of the data helps in choosing the methods.

'Stereopsis' — Prediction

Having a physical model in mind, we proceeded in two phases.

1. *Classify* data into 10 distance classes. Each class corresponds to one 10×10 surface in the grid. Distances (training target values) within each class are close to each other.
   - This part seemed trivial. We used a linear transformation to reduce dimensionality to 1, and used 9 threshold values to tell one class from another.
2. Within each class, fit a *low-order surface* to training points. The physical model guided the selection of the parametrisation of each surface.

‘Stereopsis’ — Probabilities

There are two error sources in these predictions:

1. Classification error. We assumed that the classifications were correct (when trained by only using training points, all validation samples were classified correctly and with large margins). This assumption turned out to be our fatal mistake.

2. The distance between the surface and the true target. We assumed that this error would primarily be Gaussian noise in measurements. Variance was estimated for each surface by using the training samples.

Thus, we submitted simple Gaussian predictions.

‘Stereopsis’ — Results

- *Huge* NLPD loss.
- It turned out that 499 out of 500 test samples were predicted well. 1 out of 500 samples was completely incorrect. This was a classification mistake. We obviously shouldn’t have trusted the simple classification method too much.

What else we can learn?

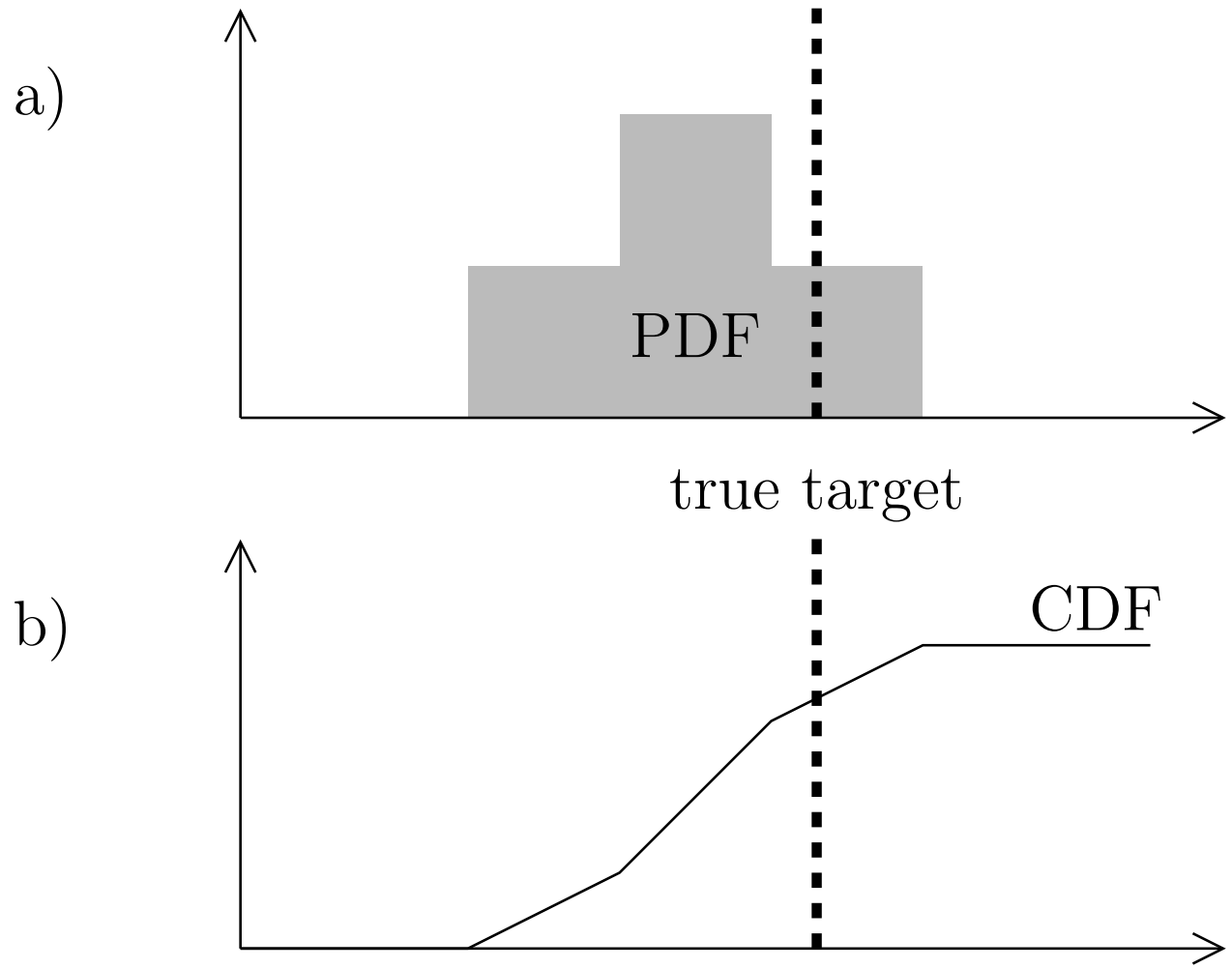
- Does the method model the expected physical phenomenon or artefacts of the calibration process?
- One needs to be careful when choosing the training and test data. E.g. random points in continuous space instead of a grid could have helped to avoid this problem.

Part 2: Scoring Predictions

Contents:

- Scoring in Challenges
- Examples of Scoring Methods: NLPD and CRPS
- Properties of Scoring Methods

Notation



Scoring in Challenges

Goals of scoring:

1. The scoring rule should encourage experts to work seriously in order to find a good method.
2. The final score should reflect how good the method is.

Indirect methods are possible, but the setting may be considerably simplified by using *proper* scoring rules. Properness means that making honest predictions is rational.

Properness is good but not enough. There are large families of proper scoring rules. Which one to choose?

Two examples follow.

Scoring Methods: NLPD and CRPS

Logarithmic score (negative log estimated predictive density, NLPD, is the corresponding *loss* function):

$$S(P, x) = \log P(x).$$

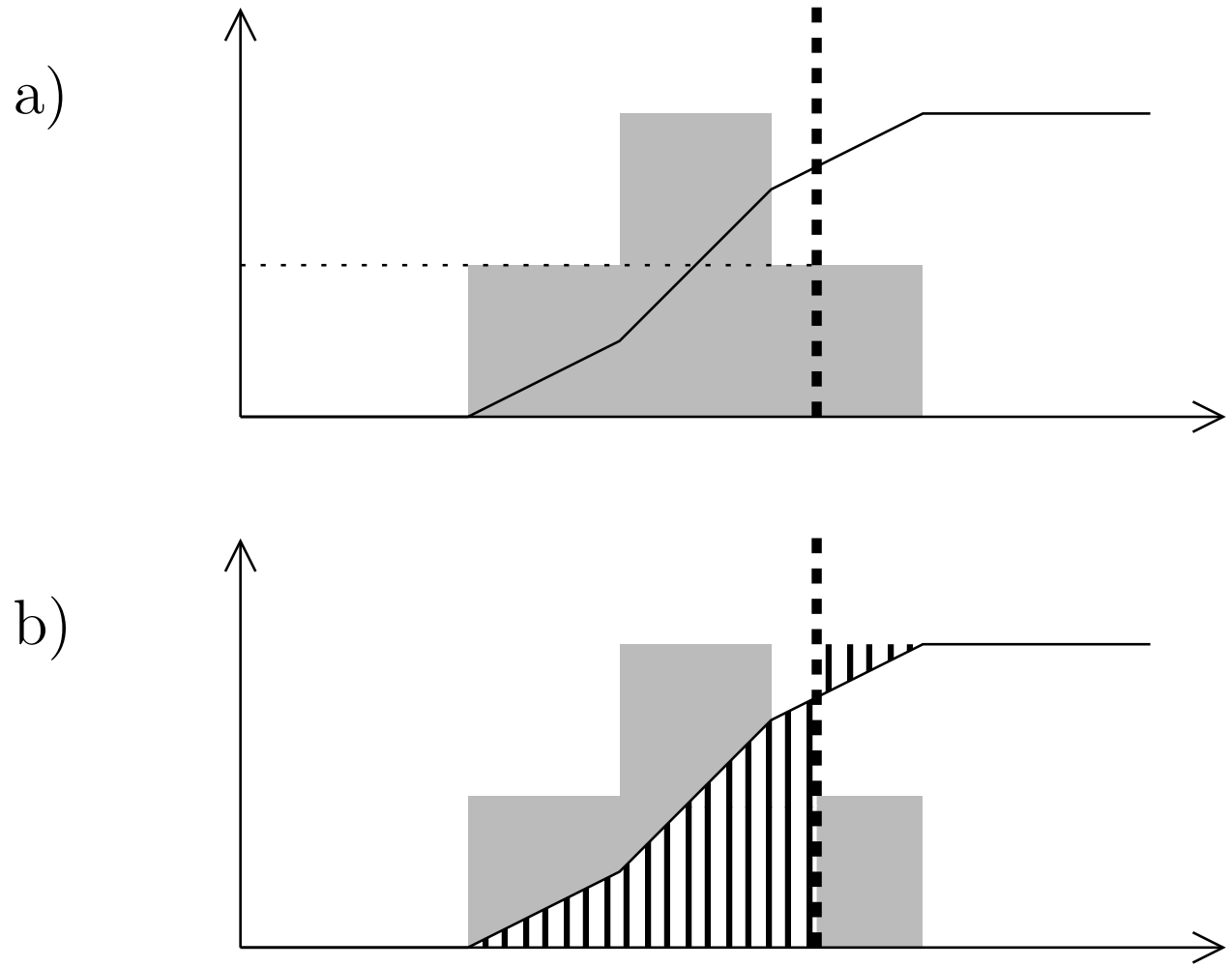
Continuous ranked probability score (CRPS):

$$S(P, x) = -\int (P(X \leq u) - R_x(X \leq u))^2 \, g(u) \, du$$

where

$$R_x(X \leq i) = 0 \quad \text{for all } i < x,$$
$$R_x(X \leq i) = 1 \quad \text{for all } i \geq x.$$

Scoring Methods: NLPD and CRPS



Scoring Methods: NLPD and CRPS

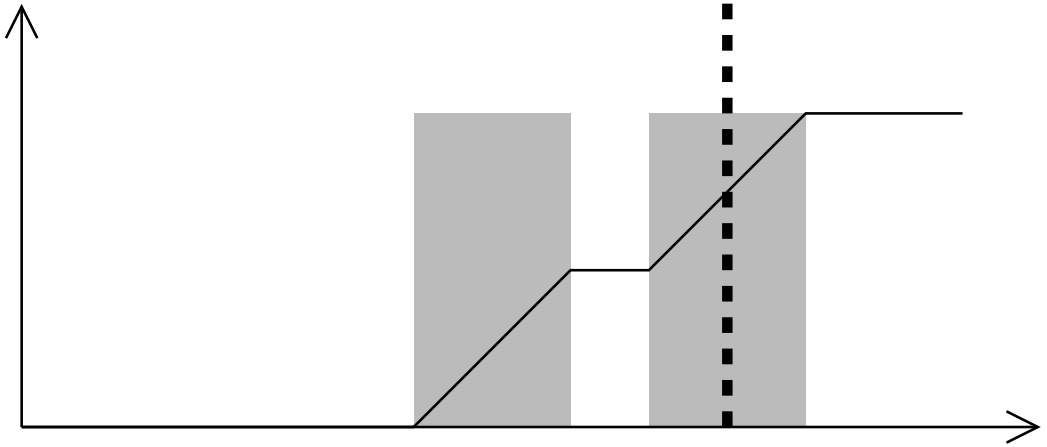
Key properties:

- NLPD is *local*, while CRPS is *distance sensitive*.
- NLPD is *not bounded*, while CRPS is always *at most* 0.

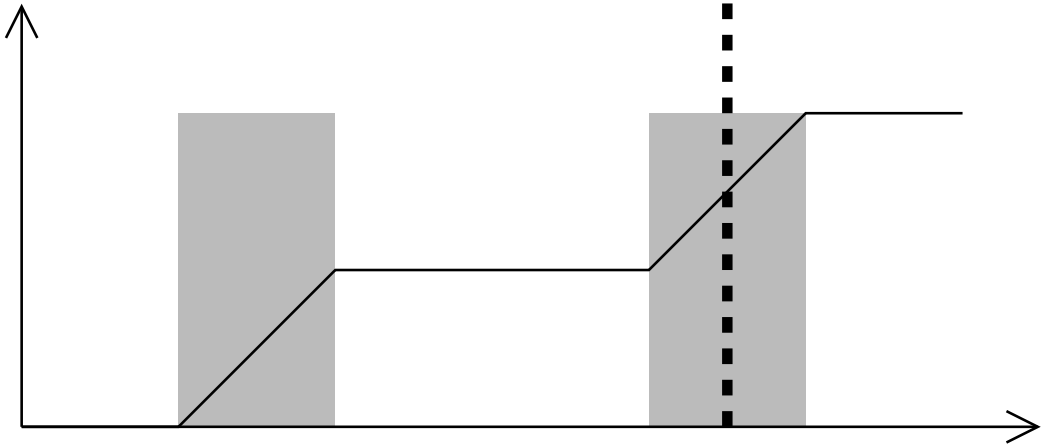
Observations follow.

Distance Sensitivity

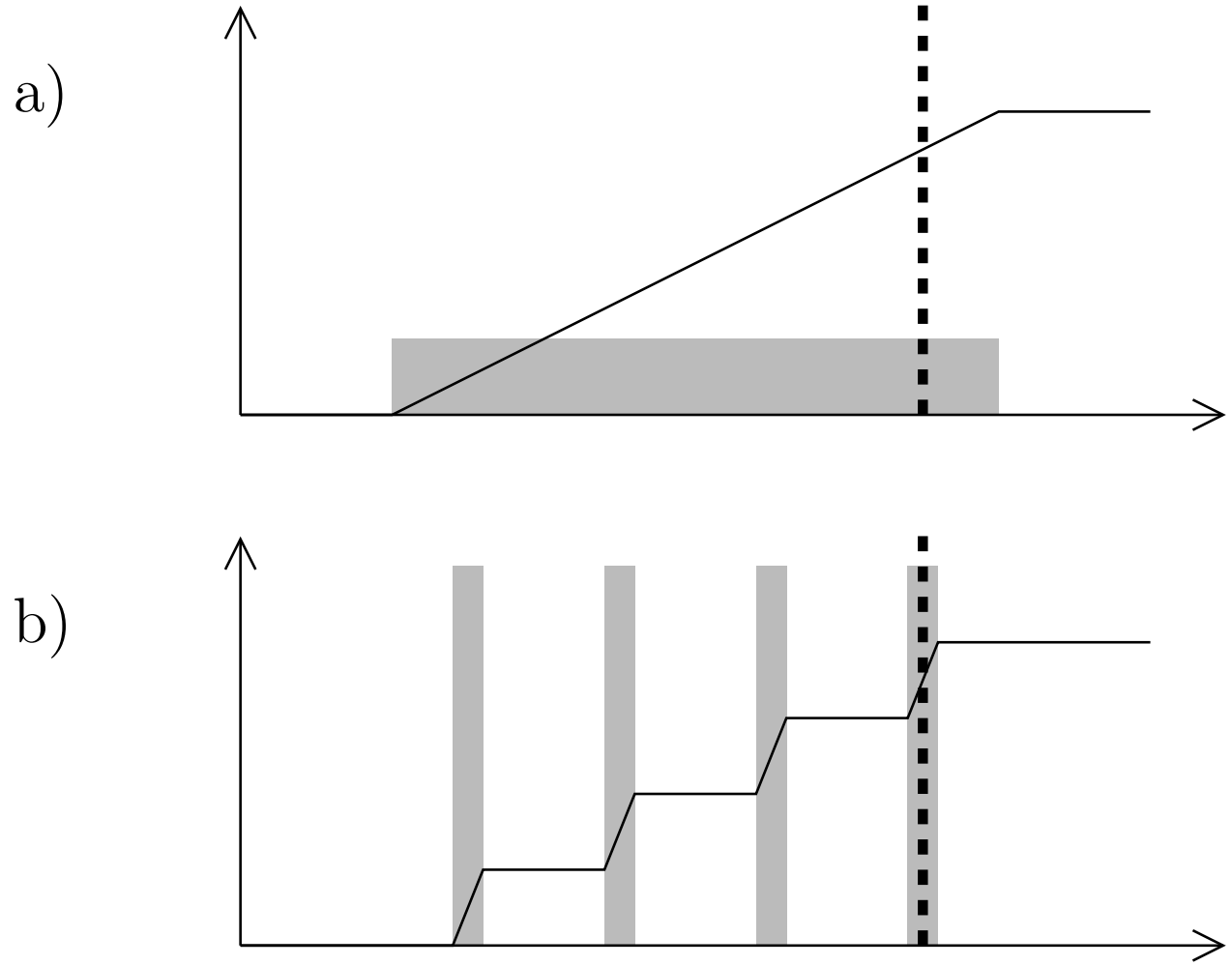
a)



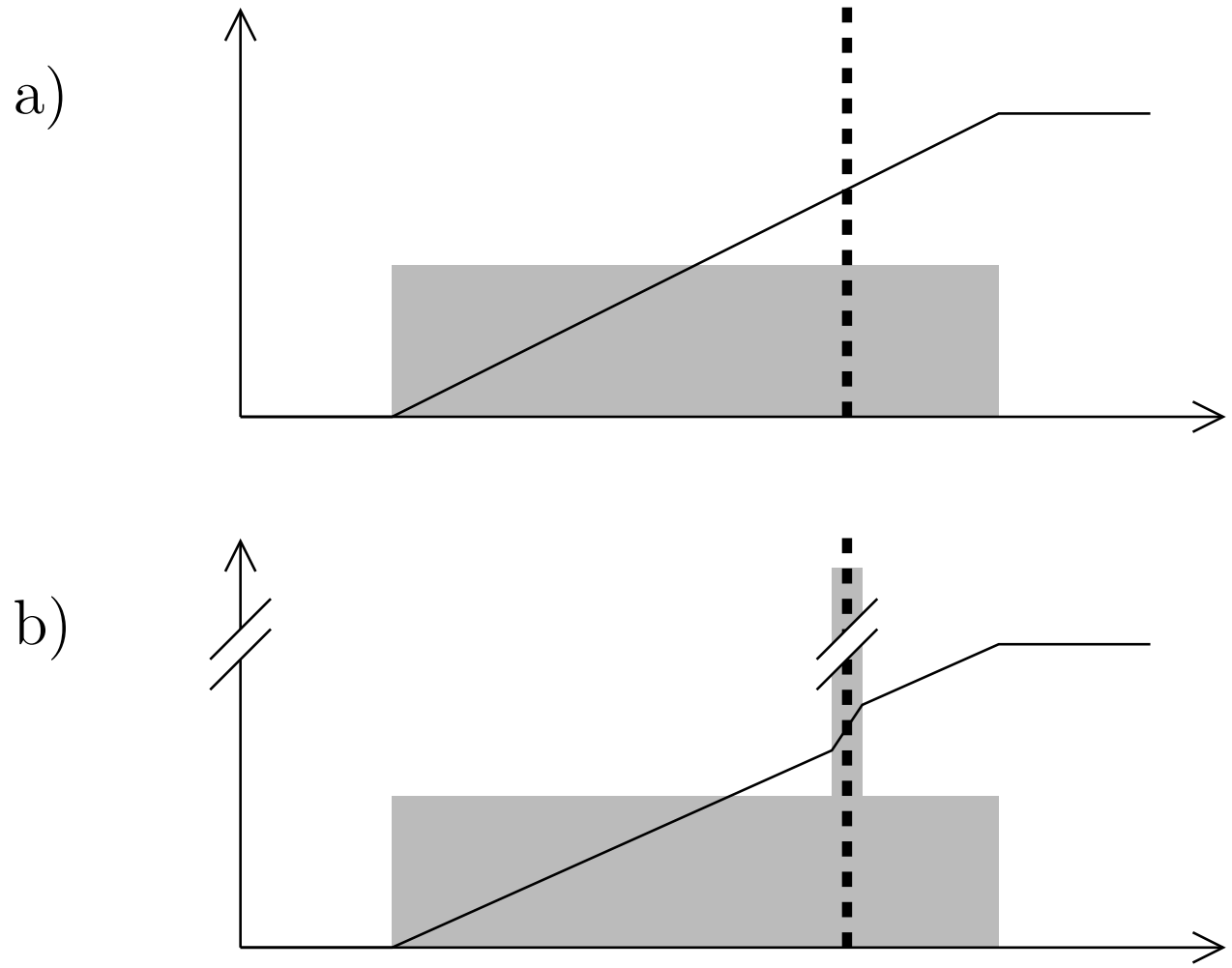
b)



Information Which Is of Little Use



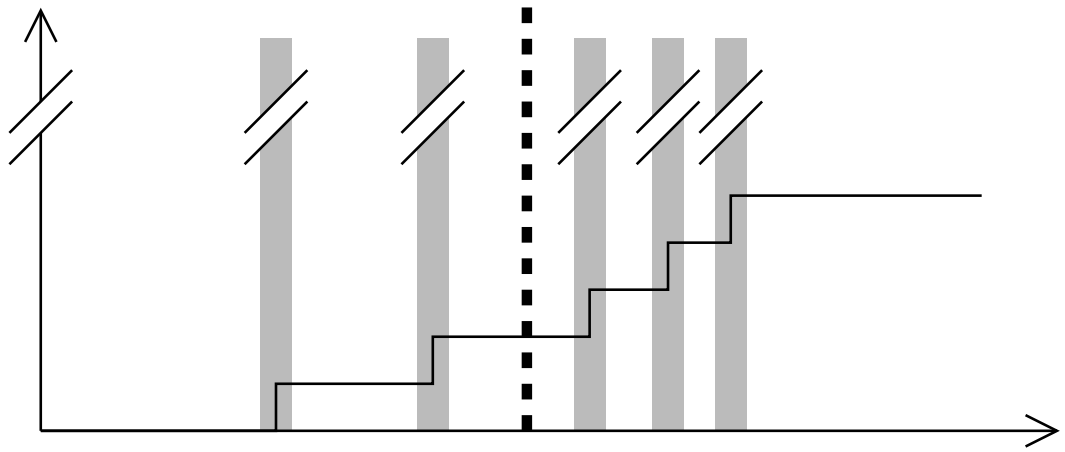
Special Values, Known Targets, etc.



Samples as Predictions

One could interpret a finite sample literally as a probability distribution with a finite set of point masses.

The NLPD loss would be meaningless. However, the CRPS score would approximate the score of the corresponding quantile prediction, but with considerably less complexity.



Summary

- We presented some methods for probabilistic prediction in regression tasks.
- There are some problems with the NLPD score. We propose using the CRPS score instead of (or in addition to) NLPD score in future challenges.
- CRPS score also makes it possible to present probabilistic predictions very easily as a finite sample.