Learning Qualitative Models from Numerical Data: Extended Abstract*

Jure Žabkar, Martin Možina, Ivan Bratko, Janez Demšar
Faculty of Computer and Information Science
University of Ljubljana, Slovenia
jure.zabkar@fri.uni-lj.si

Abstract

Qualitative models are predictive models that describe how changes in values of input variables affect the output variable in qualitative terms, *e.g.* increasing or decreasing. We describe Padé, a new method for qualitative learning which estimates partial derivatives of the target function from training data and uses them to induce qualitative models of the target function. We formulated three methods for computation of derivatives, all based on using linear regression on local neighbourhoods. The methods were empirically tested on artificial and real-world data. We also provide a case study which shows how the developed methods can be used in practice.

1 Introduction

People most often reason qualitatively. For example, playing with a simple pendulum, a five year old child discovers that the period of the pendulum increases if he uses a longer rope. Although most of us are later taught a more accurate numerical model describing the same behaviour, $T = 2\pi\sqrt{l/g}$, we keep relying on the more "operational" qualitative relation in everyday's life. Still, despite Turing's proposition that artificial intelligence should *mimic human intelligence*, not much work has been done so far in trying to learn such models from data.

We can formally describe the relation between the period of a pendulum T, its length l and the gravitational acceleration g as $T = Q(+l, -g)$, meaning that the period increases with l and decreases with g. Our definition of qualitative relatonship is based on partial derivatives: a function f is in positive (negative) qualitative relation with x over a region $\mathcal{R}$ if the partial derivative of f with respect to x is positive (negative) over the entire $\mathcal{R}$,

$$f = Q_{\mathcal{R}}(+x) \quad \equiv \quad \forall x_0 \in \mathcal{R} : \frac{\partial f}{\partial x}(x_0) > 0 \qquad (1)$$

and

$$f = Q_{\mathcal{R}}(-x) \quad \equiv \quad \forall x_0 \in \mathcal{R} : \frac{\partial f}{\partial x}(x_0) < 0. \qquad (2)$$

*This paper is an extended abstract of the AI Journal publication [Žabkar et al., 2012].

Qualitative models are predictive models that describe qualitative relations between input variables and a continuous output, for instance

$$\text{if } z > 0 \wedge x > 0 \text{ then } f = Q(+x),$$
$$\text{if } z < 0 \vee x < 0 \text{ then } f = Q(-x).$$

For the sake of clarity, we omitted specifying the region since it is obvious from the context.

In this paper we propose a new, two-step approach to induction of qualitative models from data. Let the data describe a sampled function given as a set of examples $(\mathbf{x}, y)$, where $\mathbf{x}$ are attributes and y is the function value. In the first step we estimate the partial derivative at each point covered by a learning example. We replace the value of the output y for each example with the sign of the corresponding derivative q. Each relabelled example, $(\mathbf{x}, q)$, describes the qualitative behaviour of the function at a single point. In the second step, a general-purpose machine learning algorithm is used to generalize from this relabelled data, resulting in a qualitative model describing the function's (qualitative) behaviour in the entire domain. Such models describe the relation between the output and a single input variable in dependence of other attributes.

The paper includes three major contributions: (1) the idea of transforming the problem of learning qualitative models to that of learning ordinary predictive models, (2) a new method called Padé for computing partial derivatives from data typical for machine learning, (3) an extensive experimental evaluation of the proposed setup.

The papers by Forbus [Forbus, 1984], de Kleer and Brown [de Kleer and Brown, 1984], and Kuipers [Kuipers, 1986] describe approaches that became the foundations of much of qualitative reasoning work in AI. Kalagnanam *et al.* [Kalagnanam *et al.*, 1991; Kalagnanam, 1992; Kalagnanam and Simon, 1992] contributed to the mathematical foundations of qualitative reasoning.

There are a number of approaches to qualitative system identification, also known as learning qualitative models from data. Most of this work is concerned with the learning of QDE (Qualitative Differential Equations) models, e.g. GENMODEL [Coiera, 1989; Hau and Coiera, 1997], MISQ [Ramachandran *et al.*, 1994; Richards *et al.*, 1992] and QSI [Say and Kuru, 1996]. Similarly, a general purpose ILP system (Inductive Logic Programming) to induce a model from qualita-

tive behaviours [Bratko *et al.*, 1991; Coghill and King, 2002; Coghill *et al.*, 2008].

The QUIN algorithm [Šuc and Bratko, 2001; Šuc, 2003; Bratko and Šuc, 2003] is the most relevant to the present paper. QUIN induces qualitative trees by computing the qualitative change vectors between all pairs of points in the data and then recursively splitting the space into regions which share common qualitative properties. Despite similarities, Padé and QUIN are significantly different in that Padé acts as a preprocessor of numerical data and can be used in combination with any attribute-value learning system. Padé computes qualitative partial derivatives in all example data points, and these derivatives become class values for the subsequent learning.

We overview the related work extensively in [Žabkar *et al.*, 2011].

2 Computation of Partial Derivatives

We will denote a learning example as $(\mathbf{x}, y)$, where $\mathbf{x} = (x_1, x_2, \ldots x_n)$ and y is the value of the unknown sampled function, $y = f(\mathbf{x})$.

We will introduce three methods for estimation of partial derivative of f at point $\mathbf{x_0}$. The simplest one assumes that the function is linear in a small hyper-sphere around $\mathbf{x_0}$ (Figure 1a). It computes *locally weighted linear regression* on examples lying in the hyper-sphere and considers the computed coefficients as partial derivatives. The second method, τ*-regression*, computes a single partial derivative at a time. To avoid the influence of other arguments of the function, it considers only those points in the sphere which lie in a hyper-tube along the axis of differentiation (Figure 1b). The derivative can then be computed with weighted univariate regression. Finally, the *parallel pairs* method replaces the single hyper-tube with a set of pairs aligned with the axis of differentiation (Figure 1c), which allows it to focus on the direction of differentiation without decreasing the number of examples considered in the computation.

2.1 Locally weighted regression

Let $\mathcal{N}(\mathbf{x_0})$ be a set of examples $(\mathbf{x_m}, y_m)$ such that $x_{mi} \approx x_{0i}$ for all i (Figure 1a). According to Taylor's theorem, a differentiable function is approximately linear in a neighbourhood of $\mathbf{x_0}$,

$$f(\mathbf{x_m}) = f(\mathbf{x_0}) + \nabla f(\mathbf{x_0}) \cdot (\mathbf{x_m} - \mathbf{x_0}) + R_2. \quad (3)$$

Our task is to find the vector of partial derivatives, $\nabla f(\mathbf{x_0})$. We can solve this as a linear regression problem by rephrasing (3) as a linear model

$$y_m = \beta_0 + \boldsymbol{\beta}^{\mathrm{T}}(\mathbf{x_m} - \mathbf{x_0}) + \epsilon_m, \quad (\mathbf{x_m}, y_m) \in \mathcal{N}(\mathbf{x_0}), \quad (4)$$

where the task is to find $\boldsymbol{\beta}$ (and β_0) with the minimal sum of squared errors ϵ_m over $\mathcal{N}(\mathbf{x_0})$. The error term ϵ_m covers the remainder of the Taylor expansion, R_2, as well as noise in the data.

The size of the neighbourhood $\mathcal{N}(\mathbf{x_0})$ should reflect the density of examples and the amplitude of noise. Instead of setting a predefined radius (*e.g.* $||\mathbf{x_m} - \mathbf{x_0}|| < \delta$), we consider a neighbourhood of k nearest examples and weigh the points according to their distance from $\mathbf{x_0}$,

$$w_m = e^{-||\mathbf{x_m} - \mathbf{x_0}||^2/\sigma^2}. \quad (5)$$

The parameter σ^2 is fitted so that the farthest example has a negligible weight of 0.001.

This transforms the problem into locally weighted regression (LWR) [Atkeson *et al.*, 1997], where the regression coefficients represent partial derivatives,

$$\begin{bmatrix} \beta_0 \\ \boldsymbol{\beta} \end{bmatrix} = (X^{\mathrm{T}} W X)^{-1} X^{\mathrm{T}} W Y, \quad (6)$$

where

$$X = \begin{bmatrix} 1 & x_{11} & \ldots & x_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{k1} & \ldots & x_{kn} \end{bmatrix} \quad W = \begin{bmatrix} w_1 & 0 & \cdots \\ 0 & \ddots & \\ \vdots & & w_k \end{bmatrix} \quad (7)$$

$$Y = [y_1 \ \ldots \ y_k]^T$$

The computed $\boldsymbol{\beta}$ estimates the vector of partial derivatives $\nabla f(\mathbf{x_0})$.

As usual in linear regression, the inverse in (6) can be replaced by pseudo-inverse to increase the stability of the method.

2.2 τ-regression

The τ-regression algorithm differs from LWR in the shape of the neighbourhood of the reference point. It starts with κ examples in a hyper-sphere, which is generally larger than that for LWR, but then keeps only k examples that lie closest to the axis of differentiation (Figure 1b). Let us assume without loss of generality that we compute the derivative with regard to the first argument x_1. The neighbourhood $\mathcal{N}(\mathbf{x_0})$ thus contains k examples with the smallest distance $||\mathbf{x_m} - \mathbf{x_0}||_{\setminus 1}$ chosen from the κ examples with the smallest distance $||\mathbf{x_m} - \mathbf{x_0}||$, where $||\cdot||_{\setminus i}$ represents the distance computed over all dimensions except the i-th.

With a suitable selection of κ and k, we can assume $x_{m1} - x_{01} \gg x_{mi} - x_{0i}$ for all $i > 1$ for most examples $\mathbf{x_m}$. If we also assume that partial derivatives with regard to different arguments are comparable in size, we get $\partial f/\partial x_1 (x_{m1} - x_{01}) \gg \partial f/\partial x_i (x_{mi} - x_{0i})$ for $i > 1$. We can thus omit all dimensions but the first from the scalar product in (3):

$$f(\mathbf{x_m}) = f(\mathbf{x_0}) + \frac{\partial f}{\partial x_1}(\mathbf{x_0})(x_{m1} - x_{01}) + R_2 \quad (8)$$

for $(\mathbf{x_m}, y_m) \in \mathcal{N}(\mathbf{x_0})$. We again set up a linear model,

$$y_m = \beta_0 + \beta_1 (x_{m1} - x_{01}) + \epsilon_m, \quad (9)$$

where β_1 approximates the derivative $\frac{\partial f}{\partial x_1}(\mathbf{x_0})$. The task is to find the value of β_1 which minimizes error over $\mathcal{N}(\mathbf{x_0})$.

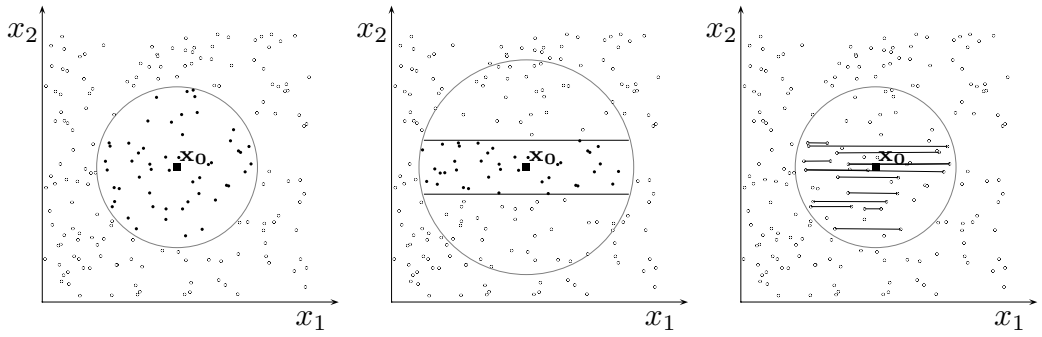
Examples in $\mathcal{N}(\mathbf{x_0})$ are weighted according to their distance from $\mathbf{x_0}$ along the axis of differentiation,

$$w_m = e^{-(x_{m1} - x_{01})^2/\sigma^2}, \quad (10)$$

where σ is again set so that the farthest example has a weight of 0.001.

The described linear model can be solved by weighted univariate linear regression over the neighbourhood $\mathcal{N}(\mathbf{x_0})$,

$$\frac{\partial f}{\partial x_1}(\mathbf{x_0}) = \beta_1 = \frac{\sum_{x_m \in \mathcal{N}(\mathbf{x_0})} w_m x_{m1} y_m}{\sum_{x_m \in \mathcal{N}(\mathbf{x_0})} w_m x_{m1}^2}. \quad (11)$$



(a) (b) (c)

Figure 1: The neighbourhoods for locally weighted regression, τ-regression and parallel pairs.

2.3 Parallel pairs

Consider two examples $(\mathbf{x_m}, y_m)$ and $(\mathbf{x_l}, y_l)$ which are close to $\mathbf{x_0}$ and aligned with the x_1 axis, $||x_m - x_l|| \approx |x_{m1} - x_{l1}|$. Under these premises we can suppose that both examples correspond to the same linear model (9) with the same coefficients β_0 and β_1. Subtracting (9) for y_m and y_l gives

$$y_m - y_l = \beta_1(x_{m1} - x_{l1}) + (\epsilon_m - \epsilon_l) \tag{12}$$

or

$$y_{(m,l)} = \beta_1 x_{(m,l)1} + \epsilon_{(m,l)}, \tag{13}$$

where $y_{(m,l)} = y_m - y_l$ and $x_{(m,l)1} = x_{m1} - y_{l1}$. The difference $y_m - y_l$ is therefore linear with the difference in the attribute values x_{m1} and x_{l1}

Coefficient β_1 again approximates the derivative $\frac{\partial f}{\partial x_1}(\mathbf{x_0})$. Note that the model has no intercept term, β_0.

To compute the derivative using (12) we take the spherical neighbourhood like the one from the first method, LWR. For each pair we compute its alignment with the x_1 axis using a scalar product with the base vector $\mathbf{e_1}$,

$$\alpha_{(m,l)} = \left| \frac{(\mathbf{x_m} - \mathbf{x_l})^{\mathrm{T}} \mathbf{e_1}}{||\mathbf{x_m} - \mathbf{x_l}||\ ||\mathbf{e_1}||} \right| = \frac{|x_{m1} - x_{l1}|}{||\mathbf{x_m} - \mathbf{x_l}||} \tag{14}$$

We select the k best aligned pairs from κ points in the hypersphere around $\mathbf{x_0}$ (Figure 1c) and assign them weights corresponding to the alignment,

$$w_{(m,l)} = e^{-\alpha^2_{(m,l)}/\sigma^2}, \tag{15}$$

with σ^2 set so that the smallest weight equals 0.001.

The derivative is again computed using univariate linear regression, $\frac{\partial f}{\partial x_1}(\mathbf{x_0}) = \beta_1 =$

$$= \frac{\sum_{(x_m,x_l)\in\mathcal{N}(\mathbf{x_0})} w_{(m,l)}(x_{m1} - x_{l1})(y_m - y_l)}{\sum_{(x_m,x_l)\in\mathcal{N}(\mathbf{x_0})} w_{(m,l)}(x_{m1} - x_{l1})^2} \tag{16}$$

3 Experiments

We present the experiments on artificially constructed data sets to test Padé with respect to accuracy and the effect of noise in the data. Further experiments are described in [Žabkar *et al.*, 2011].

			τ-regression			
	LWR	Pairs	$\kappa = 30$	50	70	100
$k = 10$	.991	.986	.948	.968	.971	.980
20	.993	.992	.929	.960	.972	.981
30	.992	.992	.909	.953	.969	.978
40	.993	.993		.950	.964	.974
50	.994	.993		.935	.961	.972

(a) Accuracy of qualitative derivatives.

			τ-regression			
	LWR	Pairs	$\kappa = 30$	50	70	100
$k = 10$	.997	.993	.990	.993	.990	.991
20	.996	.995	.983	.991	.986	.991
30	.996	.994	.983	.987	.988	.993
40	.995	.995		.978	.978	.990
50	.998	.995		.977	.987	.991

(b) Accuracy of qualitative models induced by C4.5.

Table 1: Results of experiments with $f(x, y) = x^2 - y^2$.

To assess the accuracy of induced models, we compute the derivatives and the model from the entire data set. We then check whether the predictions of the model match the analytically computed partial derivatives. We define the accuracy of Padé as the proportion of examples with correctly predicted qualitative partial derivatives. Note that this procedure does not require separate training and testing data set since the correct answer with which the prediction is compared is not used in induction of the model. Where not stated otherwise, experimental results represent averages of ten trials. For LWR, we used ridge regression to compute the ill-posed inverse in (6).

We performed experiments with three mathematical functions: $f(x, y) = x^2 - y^2$, $f(x, y) = x^3 - y$, $f(x, y) = \sin x \sin y$. We sampled them uniform randomly in 1000 points in the range $[-10, 10] \times [-10, 10]$.

Function $f(x, y) = x^2 - y^2$ is a standard test function often used in [Šuc, 2003]. Its partial derivative w.r.t. x is $\partial f/\partial x = 2x$, so $f = Q(+x)$ if $x > 0$ and $f = Q(-x)$ if $x < 0$. Since the function's behaviour with respect to y is similar, we observed only results for $\partial f/\partial x$. The accuracy of all methods is close to 100% (Table 1). Changing the values of parameters

	LWR	Pairs	τ-regression $\kappa = 30$	50	70	100
$k = 10$	.727	.690	.597	.626	.639	.647
20	.725	.792	.576	.593	.619	.646
30	.751	.879	.545	.571	.609	.618
40	.740	.919		.556	.588	.613
50	.725	.966		.541	.558	.612

(a) Accuracy of qualitative derivatives.

	LWR	Pairs	τ-regression $\kappa = 30$	50	70	100
$k = 10$	1.00	.898	.737	.880	.890	.864
20	1.00	.930	.745	.743	.811	.860
30	.978	.954	.752	.599	.813	.777
40	.971	.964		.780	.732	.700
50	.956	.986		.906	.805	.760

(b) Accuracy of qualitative models induced by C4.5.

Table 2: Results of experiments with $f(x, y) = x^3 - y$.

	LWR	Pairs	τ-regression $\kappa = 30$	50	70	100
$k = 10$	.882	.858	.863	.885	.890	.886
20	.870	.841	.820	.865	.880	.882
30	.862	.823	.769	.837	.861	.877
40	.844	.796		.799	.839	.853
50	.814	.754		.717	.810	.845

(a) Accuracy of qualitative derivatives.

	LWR	Pairs	τ-regression $\kappa = 30$	50	70	100
$k = 10$	.509	.519	.516	.512	.510	.509
20	.515	.531	.509	.518	.522	.510
30	.515	.523	.510	.513	.514	.515
40	.521	.555		.511	.507	.511
50	.516	.583		.507	.508	.515

(b) Accuracy of qualitative models induced by C4.5.

Table 3: Results of experiments with $f(x, y) = \sin x \sin y$.

has no major effect except for τ regression, where short ($\kappa = 30$ and $\kappa = 50$) and wide ($k \geq 10$) tubes give better accuracy while for very long tubes ($\kappa = 100$) the accuracy decreases with k. The latter can indicate that longer tubes reach across the boundary between the positive and negative values of x. Induced tree models have the same high accuracy.

Function $f(x, y) = x^3 - y$ is globally monotone, increasing in x and decreasing in y in the whole region. The function is interesting because its much stronger dependency on x can obscure the role of y. All methods have a 100% accuracy with regard to x. Prediction of function's behaviour w.r.t. y proves to be difficult: the accuracy of τ-regression is 50–60% and the accuracy of LWR is just over 70% (Table 2). Parallel pairs seem undisturbed by the influence of x and estimate the sign of $\partial f/\partial y$ with accuracy of more than 95% with proper parameter settings.

An interesting observation here is that the accuracy of induced qualitative tree models highly exceeds that of pointwise partial derivatives. For instance, qualitative models for derivatives by LWR reach 95–100% despite the low, 70% accuracy of estimates of the derivative. When generalizing from labels denoting qualitative derivatives, incorrect labels are scattered randomly enough that C4.5 recognizes them as noise and induces a tree with a single node.

Function $f(x, y) = \sin x \sin y$ has partial derivatives $\partial f/\partial x = \cos x \sin y$ and $\partial f/\partial y = \cos y \sin x$, which change their signs multiple times in the observed region. The accuracy of all methods is mostly between 80 and 90 percent, degrading with larger neighbourhoods (Table 3). However, the accuracy of C4.5 barely exceeds 50% which we would get by making random predictions. Rather than a limitation of Padé, this shows the (expected) inability of C4.5 to learn this checkboard-like concept.

Finally, we add various amounts of noise to the function value. The target function is $f(x, y) = x^2 - y^2$ defined on $[-10, 10] \times [-10, 10]$ which puts f in $[-100, 100]$. We added Gaussian noise with a mean of 0 and variance 0, 10, 30, and 50, *i.e.* from no noise to the noise in the range comparable to the signal itself. We measured the accuracy of derivatives and models induced by C4.5. Since the data contains noise, we set the C4.5's parameter m (minimal number of examples in a leaf) to 10% of the examples of our data set, $m = 100$. We repeated the experiment with each amount of noise 100 times and computed the average accuracies.

	$\sigma = 0$	$\sigma = 10$	$\sigma = 30$	$\sigma = 50$
LWR	.993	.962	.878	.795
τ-regression	.981	.945	.848	.760
parallel pairs	.992	.924	.771	.680

(a) Correctness of computed derivatives

	$\sigma = 0$	$\sigma = 10$	$\sigma = 30$	$\sigma = 50$
LWR	.996	.978	.956	.922
τ-regression	.991	.976	.949	.917
parallel pairs	.995	.966	.949	.901

(b) Correctness of qualitative models induced by C4.5

Table 4: Effect of noise on the accuracy of computed qualitative partial derivatives for $f(x, y) = x^2 - y^2$ with random noise from $N(0, \sigma)$.

The results are shown in Table 4. Padé is quite robust despite the huge amount of noise. As in other experiments on artificial data sets, we again observed that C4.5 is able to learn almost perfect models despite the drop in correctness of derivatives at higher noise levels.

4 Conclusion

We introduced a new approach to learning qualitative models which differs from existing approaches by its trick of translating the learning problem into a classification problem and then applying the general-purpose learning methods to solve it. We mostly explored the first step that involves the estimation of partial derivatives. The second step opens a number of other interesting research problems, which we leave open for further research in the area.

References

[Atkeson *et al.*, 1997] C. Atkeson, A. Moore, and S. Schaal. Locally weighted learning. *Artificial Intelligence Review*, 11:11–73, 1997.

[Bratko and Šuc, 2003] I. Bratko and D. Šuc. Learning qualitative models. *AI Magazine*, 24(4):107–119, 2003.

[Bratko *et al.*, 1991] I. Bratko, S. Muggleton, and A. Varšek. Learning qualitative models of dynamic systems. In *Proc. Inductive Logic Programming ILP-91*, pages 207–224, 1991.

[Coghill and King, 2002] S. M.; Coghill, G. M.; Garrett and R. D. King. Learning qualitative models in the presence of noise. In *Proc. Qualitative Reasoning Workshop QR'02*, 2002.

[Coghill *et al.*, 2008] George M. Coghill, Ashwin Srinivasan, and Ross D. King. Qualitative system identification from imperfect data. *J. Artif. Int. Res.*, 32(1):825–877, 2008.

[Coiera, 1989] E. Coiera. Generating qualitative models from example behaviours. Technical Report 8901, University of New South Wales, 1989.

[de Kleer and Brown, 1984] J. de Kleer and J. Brown. A qualitative physics based on confluences. *Artificial Intelligence*, 24:7–83, 1984.

[Forbus, 1984] K. Forbus. Qualitative process theory. *Artificial Intelligence*, 24:85–168, 1984.

[Hau and Coiera, 1997] D. Hau and E. Coiera. Learning qualitative models of dynamic systems. *Machine Learning Journal*, 26:177–211, 1997.

[Kalagnanam and Simon, 1992] Jayant Kalagnanam and Herbert A. Simon. Directions for qualitative reasoning. *Computational Intelligence*, 8(2):308–315, 1992.

[Kalagnanam *et al.*, 1991] Jayant Kalagnanam, Herbert A. Simon, and Yumi Iwasaki. The mathematical bases for qualitative reasoning. *IEEE Intelligent Systems*, 6(2):11–19, 1991.

[Kalagnanam, 1992] Jayant Ramarao Kalagnanam. *Qualitative analysis of system behaviour*. PhD thesis, Pittsburgh, PA, USA, 1992.

[Kuipers, 1986] Benjamin Kuipers. Qualitative simulation. *Artificial Intelligence*, 29:289–338, 1986.

[Ramachandran *et al.*, 1994] S. Ramachandran, R. Mooney, and B. Kuipers. Learning qualitative models for systems with multiple operating regions. In *Working Papers of the 8th International Workshop on Qualitative Reasoning about Physical Systems*, Japan, 1994.

[Richards *et al.*, 1992] B. Richards, I. Kraan, and B. Kuipers. Automatic abduction of qualitative models. In *Proceedings of the National Conference on Artificial Intelligence*. AAAI/MIT Press, 1992.

[Say and Kuru, 1996] A. Say and S. Kuru. Qualitative system identification: deriving structure from behavior. *Artificial Intelligence*, 83:75–141, 1996.

[Šuc and Bratko, 2001] D. Šuc and I. Bratko. Induction of qualitative trees. In L. De Raedt and P. Flach, editors, *Proceedings of the 12th European Conference on Machine Learning*, pages 442–453. Springer, 2001. Freiburg, Germany.

[Šuc, 2003] D. Šuc. *Machine Reconstruction of Human Control Strategies*, volume 99 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, Amsterdam, The Netherlands, 2003.

[Žabkar *et al.*, 2011] Jure Žabkar, Martin Možina, Ivan Bratko, and Janez Demšar. Learning qualitative models from numerical data. *Artificial Intelligence*, 175(9 - 10):1604 – 1619, 2011.