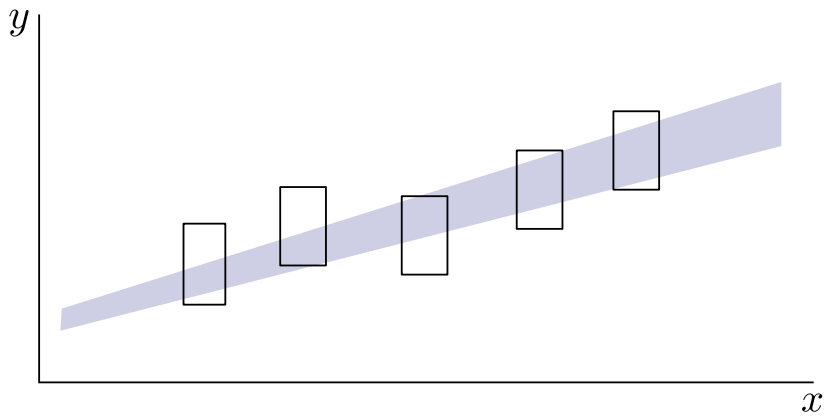

Figure 9.7: An example of $\boldsymbol{r} = \boldsymbol{p}_1 x + \boldsymbol{p}_2$. The band actually forms an interval line, which passes through each interval box.

The resulting $\boldsymbol{r}$ can be viewed as a multi-dimensional band. A two-dimensional example can be found in Figure 9.7.

As it was explained, there are various types of interval regression. They vary in computation of interval parameters $\boldsymbol{p}$. For example, $\boldsymbol{p}$ could be computed in such a way to force the band $\boldsymbol{r}$ to contain all the data tuples, or at least to cross all the interval data. For our purpose the interval least squares approach is the most meaningful.

Definition 9.3. For a given data: an $m \times n$ interval matrix $\boldsymbol{X}$, where its ith row is the tuple

$$(\boldsymbol{x}_1^i, \boldsymbol{x}_2^i, \ldots, \boldsymbol{x}_n^i),$$

and an m-dimensional column vector $\boldsymbol{y}$, where its coefficients are $\boldsymbol{y}^i$, the interval parameters $\boldsymbol{p}$ of the interval least squares estimation are defined in the following way,

$$\boldsymbol{p} = \square\{p : X^T X p = X^T y \text{ for some } X \in \boldsymbol{X}, y \in \boldsymbol{y}\}.$$

In Section 7.2.4 we addressed how to solve such a problem. We basically solved the following system

$$\begin{pmatrix} I & \boldsymbol{X} \\ \boldsymbol{X}^\top & 0 \end{pmatrix} \begin{pmatrix} p' \\ p \end{pmatrix} = \begin{pmatrix} \boldsymbol{y} \\ 0 \end{pmatrix} \tag{9.1}$$

using the means of some method for solving square interval systems from Chapter 5. The last n coefficients of the resulting enclosure give an enclosure on $\boldsymbol{p}$.

When we take a look at $(\boldsymbol{X}, \boldsymbol{y})$ data obtained by MBW procedure in Figure 9.6 we realize that

- $X = \boldsymbol{X}$ is thin, it consist of integers only (numbers of breaths) – we use such a form to avoid using intervals on the x-axis,
- intervals are only at the right-hand side $\boldsymbol{y}$,

- We want to use regression with nonlinear models that are linearizable, therefore $X^\top X$ is going to be small $n \times n$, $(n = 2, 3, 4)$, depending on the number of parameters of the model used (see the table 9.4 in advance),
- $X, \boldsymbol{y} > 0$ (component-wise).

Using these favorable properties, we hoped to design a method returning tighter enclosures than (9.1). Unfortunately, we were not able to find such a method. We believe that it is a really hard task since the mentioned properties are also in favor of (9.1). However, we were able to rewrite the formulas to obtain algorithms that are much faster.

9.7.1 Case 2×2

When the matrix X is of size $m \times 2$ (the left column is consists of ones, and the right one of numbers $1, \ldots, m$), then $X^T X$ is of size 2×2. We can apply the state of the art supersquare approach, however, in this case the "not recommended" approach of solving the interval normal equation $X^T X p = X^T \boldsymbol{y}$ will pay off. This actually means computing an enclosure of p as

$$\boldsymbol{p} = \left((X^T X)^{-1} X^T\right) \boldsymbol{b}. \tag{9.2}$$

When computing an inverse matrix, fractions can occur and therefore so can machine nonrepresentable numbers. That is why, we need to compute in a verified way with intervals. Nevertheless, it is advantageous to postpone the interval computation as far as possible, because the classical arithmetic is usually faster (e.g., in Octave or Matlab). In this case we use the simple shape of the 2×2 matrix inverse

$$(X^T X)^{-1} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{ad - bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}.$$

It is possible to compute $X^T X$ in floating point arithmetics since X contains only integers; similarly for $ad - bc$.

When computing the expression $(X^T X)^{-1} X^T \boldsymbol{y}$, $\boldsymbol{y}$ is multiplied by an interval matrix, this unfortunately causes large growth of interval radii. And then it is multiplied again with the matrix $(X^T X)^{-1}$ which causes another growth. More suitable way is to rearrange the expression to multiply the integer parts (matrices) first and then multiplying with the interval elements. Thus, the enclosure of $\boldsymbol{p}$ can be computed as

$$(M X^T)(\boldsymbol{q}\boldsymbol{y}),$$

where

$$M = \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}, \quad \boldsymbol{q} = \square\left(\frac{1}{ad - bc}\right).$$

We tested the difference between (9.1), which was solved by HBR method (`supsq`) and (9.2) solved directly by computing the verified inverse (`normal`), and the same procedure but with postponing the interval operations (`postponed`). The differences between approaches are clearly seen in the following example.

Table 9.2: Average computation times (in seconds) for 2×2 systems (Example 9.4) for the supersquare approach (`supsq`) and solving interval normal equations without (`normal`) and with postponing the interval operations (`postponed`).

m	`supsq`	`normal`	`postponed`
50	0.224	0.067	0.013
100	0.407	0.069	0.014
150	0.609	0.070	0.014
200	0.857	0.071	0.014
250	1.156	0.070	0.014

Example 9.4. The difference was tested on random systems for sizes up to $m = 250$, which represents the ceiling for the maximum number of breaths generally occurring during MBW testing. To generate a random right-hand side we first generated random intervals with centers from $[-10, 10]$ and fixed radii equal to 1 and then the intervals were placed along a random line and then shifted by a random number in $[-5, 5]$. The testing was done using LAPTOP setting. For each size we tested on 100 random systems. Both methods in all cases computed identical enclosures for $\boldsymbol{p}$. However, average computation times were different, they are displayed in the following Table 9.2.

9.7.2 Case 3×3 and larger

It would be more complicated to find similar inverse formula for a general square matrix. This time we refrain from postponing interval computations.

Example 9.5. We again compare with the supersquare approach. The test data were generated in a similar way to Example 9.4. The only difference is that the right-hand side intervals were placed along a parabola. The obtained enclosures of $\boldsymbol{p}$ are again identical and the average computation times are displayed in Table 9.3. The `normal` method is still faster than supersquare approach.

9.8 In search for a model

Inspired by Figure 9.6, the main goal is to derive the following function

$$f(n), \text{ for } n = 1, 2, \ldots$$

where n is the number of a peak (or breath), where the initial peak has number 1. The function f returns a nitrogen concentration at each peak n (it can be an interval concentration) and should plausibly model the nitrogen concentration at each peak. We call such a function f a *nitrogen washout curve model*. This goal was addressed earlier in [187] using a simplified model of lungs. They were not able to compute with

Table 9.3: Average computation times (in seconds) for 3×3 systems (Example 9.5) for the supersquare approach (`supsq`) and solving interval normal equations with (`normal`).

m	`supsq`	`normal`
50	0.26	0.07
100	0.42	0.07
150	0.67	0.07
200	0.92	0.07
250	1.19	0.07

models having more parameters due to the limited computational power (they handled many calculations manually). Their approach could be described as "bottom-up". A similar approach but for a different goal can be seen, e.g., in [212].

Our approach is slightly different, we could call it "top-down". Using a computer we explore the most frequent mathematical models of decay and test their ability to fit the measured medical data. Such a fitting might help to obtain more information about the real behavior of the nitrogen washout process and such knowledge will help to better predict the behavior of an incomplete measurement.

9.8.1 Center data

In the previous sections we showed how to derive interval data from a measured real patient data. To have at least rough idea about the behavior of the nitrogen washout process, classical least squares data fitting was applied on center data (for a while we consider only midpoints from all intervals).[1]

We are interested in fitting curves for which the process of good fitting can be transformed to solving a linear system of equations. The quality of fit was measured by rMSE which is the square root of MSE (mean squared error). We fit the data in least squares manner. If we evaluate the measurements visually, we could detect "exponential"-like decay in all data. An example could be seen in Figure 9.6. Many papers and books (also possibly the medical software shipped with the machine Exhalyzer D) describe this decay as an exponential function [32]. This is one of the classical fitting models. When talking about classical fitting models we tried to find the most suitable one among them. From the large collection of models [205] we selected the following model candidates fulfilling the visual criteria first. They are summarized in Table 9.4. For each model in the left column there is the abbreviation by which we address the model, in the second column there is the mathematical description of the model and in the third column there are the parameters that need to be computed to fit a given dataset with this model. As already mentioned, all of these models can be linearized. For a detailed description of this process for each model see [205].

[1] For this purpose we used a different data set to the one from Subsection 9.4.2. We selected cleaner data to make them more suitable for regression.

Table 9.4: Table of the fitting models used.

model	function $f(x)$	parameters
`exp`	$ae^{(bx)}$	a, b
`explin`	$a + bx + ce^x$	a, b, c
`pow`	ax^b	a, b
`exppow`	$ax^b c^x$	a, b, c
`log`	$a + b\log(x)$	a, b
`loglin`	$a + bx + c\log(x)$	a, b, c
`explin`	$a + bx + ce^x$	a, b, c
`explog`	$a + b\log(x) + ce^x$	a, b, c
`exploglin`	$a + bx + c\log(x) + de^x$	a, b, c, d

For each dataset (one measurement) each model was fitted and rMSE computed. As stated earlier, the 2.5% and 5% concentration level is significant for medical specialists. When we follow the nitrogen curve in time beyond the 2.5% level of concentration, it can be seen that the concentration peaks can be interpolated with a nearly horizontal line. It is difficult for all models to fit properly such slowly decreasing end. That is why we also measured the quality of fit to a level where something is "still happening" (the curve does not decrease so slowly) – up to 5%. The rMSE results can be seen in Tables 9.5, 9.6, 9.7 and 9.8 at the end of this chapter.

From the perspective of rMSE measure the model `loglin` is the winner. The rMSE penalizes heavily the large misfits. If we take a look at the `loglin` curve it can fit the initial part of the washout curve pretty well. All other models are penalized, except for the model `exploglin`. It sometimes seems to be better, however, the coefficient in exponential member of the formula (d) is usually an extremely tiny number ($\sim 10^{-10}$). That is why this model is usually the same as `loglin`. From the perspective of Occam's principle we further consider only the `loglin` model. However, the curve with the best rMSE fit does not have to be necessary the best for the sake of prediction of the washout curve behaviour. Notice that the model `exp`, which is often used in describing the nitrogen washout curve in medical literature, is not so accurate.

When data sets were shortened up to the point where the nitrogen concentration decreases below 5% of its initial concentration, the model `exppow` works much better on this initial phase; and its fitting error improved. Nevertheless, the best fitting model is still `loglin`. We therefore have some candidates for interval fitting models. We omit the model `exploglin`, since it is too complicated. We exclude the model `log` since it is contained in `loglin` and does not have better results than `loglin`. We also cast out models `explin` and `explog` due to a large error rate. We have four remaining candidates – `exp`, `pow`, `exppow`, `loglin` – that we further use. None of the checked model curves was able to accurately fit the data from the 5% to 2.5%. The level of 5% therefore seems to be a meaningful level that still enables possible plausible fitting with one of the classical models. This could also be an important fact for current discussions about advantages of LCI5 over LCI2.5. However, we must be careful not

to reach the conclusions too quickly, because the part of the washout curve between 5% and 2.5% can possibly contain some important information about the quality of patient airways. Tossing a terminal part of the data away might mean tossing away an important information for further medical analysis.

9.8.2 Interval models

We took the four candidates on fitting curves – `exp`, `pow`, `exppow`, `loglin` – and provided the interval fitting of each model in the least squares manner. Each fitting of a nonlinear model can be transformed to solving an interval linear system of equations (the process is thoroughly described in [24]) and then solved by the means described in Section 9.7. Unfortunately, the results were not encouraging – the resulting interval washout curve models are too wide to yield any insight on the process of nitrogen washout. Another reason for such an overestimation might be the fact that solving an interval linear system exactly is difficult and we produced only an overestimated enclosure. Also induced dependencies in the supersquare system may play an important role (see 7.2.4). Shapes typical for each interval washout model are depicted in Figure 9.8. The `exp` function misses the initial and terminal part of the washout data. The `pow` model misses the initial part. The `exppow` model is usually too wide, however, it contains the data inside the interval curve. The `loglin` model usually tends to widen in time; ruining any possibility of prediction. As shown in the next subsection, we blame the accuracy of the sensors. Hence our result, although negative, might be a serious contribution to the ongoing discussions on quality of sensors.

9.8.3 Hypothetical sensors

We showed that problem of quality of fitted interval models lies within precision of current sensors (0.3% for O_2 sensor and 5% for CO_2 sensor of Exhalyzer D machine) and also within the methods for solving interval systems of equations. One might claim that the main flaw lies in the methods for solving interval systems and their overestimation. To shed more light on this, let us assume we have sensors with a better accuracy by one order, i.e, 0.03% for O_2 sensor and 0.5% for CO_2 sensor.

Let us repeat the same procedure as in Figure 9.8, this time for the hypothetical sensors. The surprising results are displayed in Figure 9.9. We checked all the four mentioned models manually by visual evaluation. We omitted the model `pow`, because it gave poor fitting results in the initial parts. We also omitted the model `exp`. Although, it gave very narrow curves it resulted in a poor fit. We checked the two remaining models – `exppow` and `loglin`. The problems with `loglin` still persist. Even for narrow intervals the curve tends to rise at its end. This gives us the winning description model – `exppow`. If we take a look at Figure 9.9, we see that the behaviour of `exppow` model does not fit the data well under the horizontal line (5% concentration level). However, it seems to work well before it crosses the level . We further check its properties in the next section.

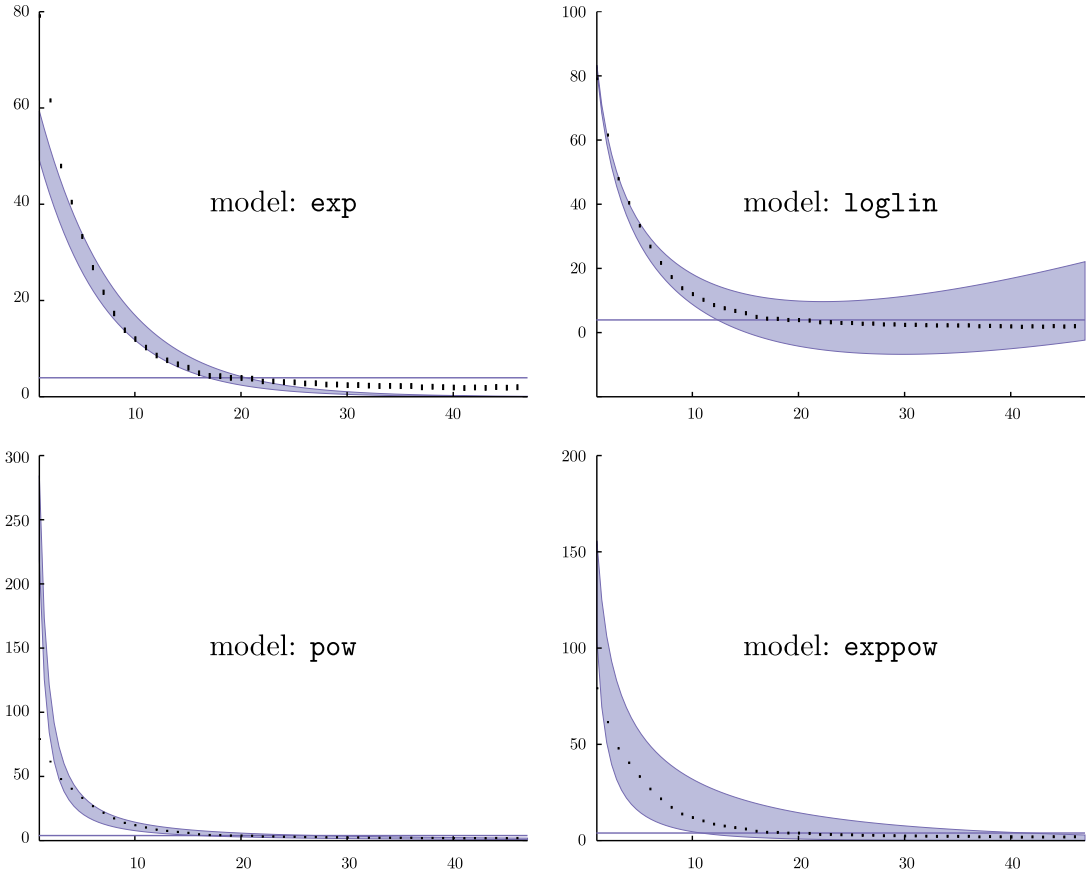

Figure 9.8: Interval curves fitting a real data with real measurement errors – typical behavior. The tiny rectangles represent the interval data. The horizontal line represents the level of 5% of the initial nitrogen concentration. Notice that the y-scale of each graph is different. The darker area corresponds to the interval least squares fitting curves (interval washout models).

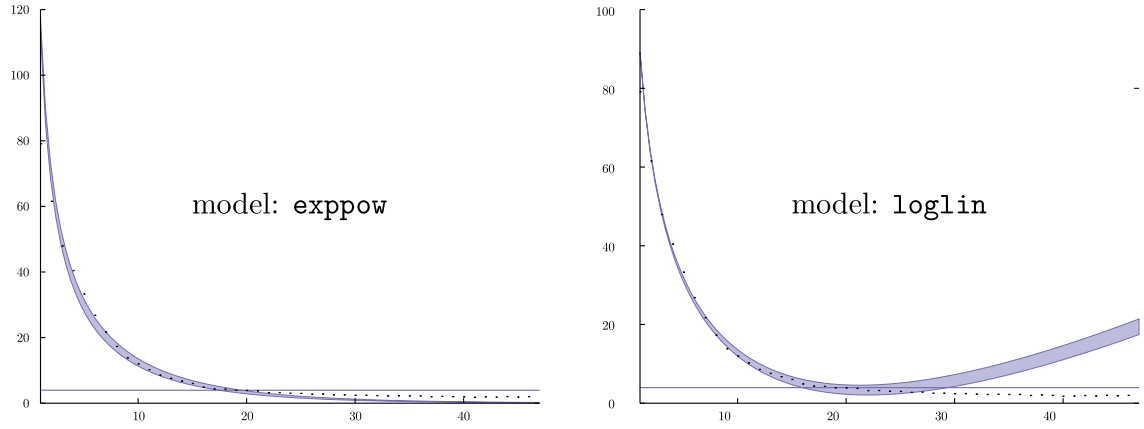

Figure 9.9: Interval curves fitting a real data with hypothetical measurement errors – typical behavior. The tiny rectangles represent the interval data. The horizontal line represents the level of 5% of the initial nitrogen concentration. Notice that the y-scale of each graph is different. The darker area corresponds to the interval least squares fitting curves (interval washout models).

9.8.4 Prediction

As it was said the level of nitrogen concentration where we stop the measurement is 2.5% or 5%. This boundary was set historically. For young uncooperative patients it might be difficult to prevent leaks and maintain calm and regular breathing for a longer period of time. Sometimes the measurement must be aborted. In order to not waste the so far good measurement we can try to predict the successive behavior of the washout curve. Using the previously developed interval washout models we focus on determination of the terminal breath of a measurement. To remind the definition, for a given level of nitrogen concentration (20%, 10%, 5% or 2.5%), the *terminal breath* for this concentration is defined to be the first one of the three consecutive breaths with concentration below the respective level.

We limited our prediction to the part of the washout curve between 10% and 5%. The goal was to predict an interval containing the terminal breath at 5% level and compare it with the real terminal breath at the corresponding level. For the prediction we used the hypothetical sensors only, the results are in Table 9.9 at the end of the chapter.

In the case of hypothetical sensors, the prediction is not generally bad. However, in some cases the prediction is completely wrong. We conclude that none of the tested models is completely suitable for absolutely correct prediction. Nevertheless, the quality of prediction brings us to the very important question we tackle more in the following subsection.

9.8.5 An alternative clinical index?

The prediction of the washout curve in current software (Spiroware) is of poor quality. We could see that the prediction using verified interval regression is also not too

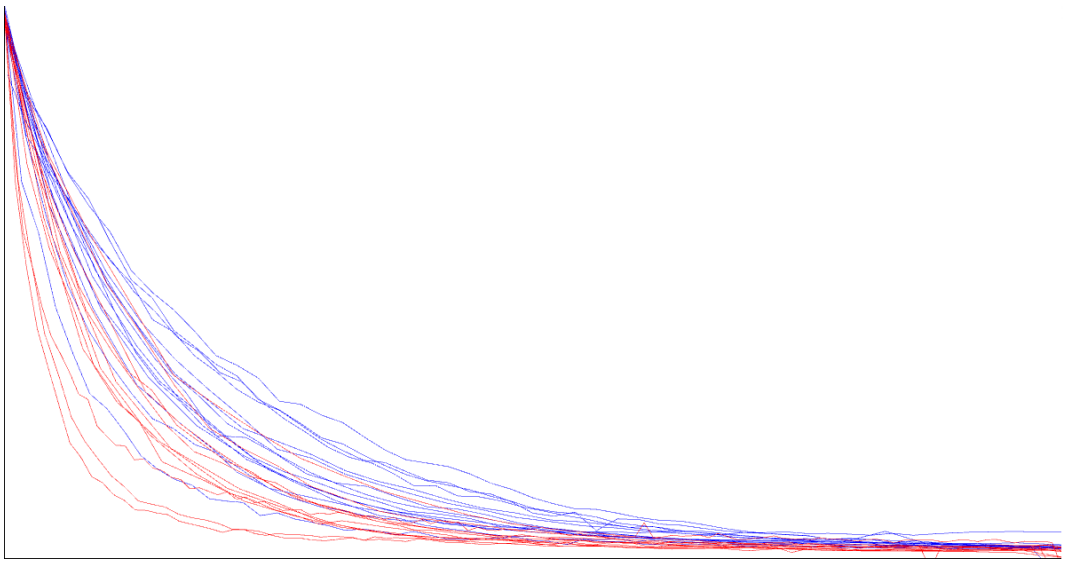

Figure 9.10: Washout curves (real data) of all patients normalized to the same length. The blue curves correspond to healthy persons, the red curves correspond to patients with cystic fibrosis.

trustworthy. The problem lies in an unsatisfactory model of the nitrogen washout process. We discussed many washout curve models, however none of them was plausible enough (for the purpose of prediction). Before starting to seek for better models, it needs to be specified, why exactly do we need predictions and models of washout process. One reason has been documented previously on an example of an interrupted measurement because of patient's weak cooperation. Indeed, the possibility to predict washout process would be of a great clinical value. Unfortunately, our results indicate, that predictions are not possible within the currently used approach to washout data analysis.

Let us say we want to predict LCI from an incomplete measurement. To derive the LCI, the FRC is also needed. For FRC derivation we need to compute V_{out} (as an integration of flow), therefore we need to know the missing flow data whose prediction is nearly impossible (too jagged shape of the flow curve). In conclusion, even if we had a good prediction of nitrogen washout behavior, there is no way to compute a meaningful LCI with this prediction.

With that a new question arises – can LCI be replaced by another index describing ventilation inhomogeneity and being more suitable to for prediction (and also robust enough to overcome some inaccuracy of prediction)? Our initial hypothesis was to use the information of curvature of the washout curve. However when the curves were normalized to stretch over the same time window, we obtained Figure 9.10. It shows that the patients cannot be simply separated as healthy or as having cystic fibrosis according to the curvature of the washout curve. Hence finding a new clinical index enabling prediction still remains a challenge.

9.9 Results relevant for medicine

We summarize the results that might be relevant to the ongoing medical discussions in the form of the following list:

- We demonstrated that the models that are usually used in literature for description of the behavior of the nitrogen washout process are not plausible.
- We showed that if we consider the classical fitting models, the best model (but still not ideal) for the washout curve description is `exppow`.
- Fitting the data with classical models up to 5% is much more achievable than the attempts to fit the data up to 2.5%.
- We gave an argument using interval analysis that current accuracy of Exhalyzer D sensors seems to be insufficient for interval data estimation and making reasonable predictions.
- If we had sensors with better accuracy just by one order the verified fitting would work.
- It is impossible to predict the future value of LCI based on an interrupted measurement due to properties of LCI.
- New clinical indices should be developed to suit prediction.
- Healthy persons and patients with cystic fibrosis cannot be simply distinguished by curvature of the washout curve.

In our work numerous ways of future research emerged – finding better models of the washout process, combination of the top-down and bottom-up approach in washout modeling, search for new clinical indices that will enable better prediction. It would be also interesting to combine the algebraic approach to uncertainty with the statistical one.

Table 9.5: Healthy persons – rMSE for fitting up to 2.5% of the initial nitrogen concentration.

No.	exp	explin	pow	exppow	log	loglin	explin	explog	exploglin	H/CF
1	7.15	23.39	28.44	6.33	4.29	**1.81**	23.39	22.54	3.50	H
2	7.98	22.22	24.45	6.59	4.64	**1.44**	22.22	21.56	2 4.59	H
3	10.67	11.84	11.30	8.11	6.26	**1.16**	11.84	5.70	6.98	H
4	10.07	11.67	11.11	5.44	6.32	**1.24**	11.67	5.75	0.96	H
5	10.80	11.82	8.80	5.94	6.63	**1.58**	11.82	6.12	1.32	H
6	8.86	10.85	16.38	5.79	4.82	**1.24**	10.85	4.68	0.77	H
7	10.71	19.15	5.91	9.44	5.71	**1.09**	19.15	19.15	19.15	H
8	7.00	10.40	21.37	4.63	4.25	**1.18**	10.40	3.83	1.18	H
9	7.10	10.60	20.71	3.81	4.43	**1.27**	10.60	4.05	1.26	H
10	3.44	23.75	31.45	2.35	2.67	**2.18**	23.75	23.75	23.75	H
11	4.27	25.36	35.95	3.04	3.03	**2.16**	25.36	25.36	25.36	H
12	3.61	24.58	33.17	2.14	2.58	**1.84**	24.58	24.58	24.58	H
13	2.74	26.09	36.08	**1.13**	2.29	1.87	26.09	26.09	26.09	H
14	5.30	10.11	24.22	2.44	3.82	**1.46**	10.11	3.40	1.45	H
15	10.30	16.14	6.34	7.50	5.96	**1.97**	16.14	16.14	16.14	H

Table 9.6: Patients with cystic fibrosis –rMSE for fitting up to 2.5% of the initial nitrogen concentration.

No.	exp	explin	pow	exppow	log	loglin	explin	explog	exploglin	H/CF
1	8.01	11.17	13.76	2.58	5.98	**1.34**	11.17	5.16	1.16	CF
2	9.54	12.69	**2.97**	4.05	6.23	3.26	12.69	12.45	12.45	CF
3	10.32	33.33	**2.71**	7.47	6.46	2.91	33.33	13.55	33.33	CF
4	9.08	11.69	11.43	3.85	7.00	**1.32**	11.69	5.78	0.81	CF
5	10.92	18.09	8.15	7.76	6.17	**2.20**	18.09	18.09	18.09	CF
6	8.31	14.40	5.36	3.38	4.69	**1.90**	14.40	14.40	14.40	CF
7	9.21	11.61	10.54	3.90	6.74	**1.27**	11.61	5.67	0.79	CF
8	7.55	11.04	22.42	5.86	4.89	**1.66**	11.04	4.35	1.66	CF
9	9.85	11.69	16.16	7.68	5.64	**1.00**	11.69	17.42	10.94	CF
10	7.38	10.78	23.83	5.41	4.53	**1.03**	10.78	4.07	1.03	CF
11	5.30	10.13	22.82	3.24	4.14	**1.35**	10.13	3.49	1.35	CF
12	7.40	10.88	19.37	4.57	5.04	**1.14**	10.88	4.43	1.11	CF

Table 9.7: Healthy persons – rMSE for fitting up to 5% of the initial nitrogen concentration.

No.	exp	explin	pow	exppow	log	loglin	explin	explog	exploglin	H/CF
1	1.99	7.80	20.35	**0.92**	2.01	1.85	7.80	1.72	1.69	H
2	3.46	8.45	18.63	1.84	2.42	**1.70**	8.45	1.96	1.59	H
3	4.51	8.83	14.25	2.22	3.57	**1.04**	8.83	2.51	0.98	H
4	5.67	9.27	11.05	1.83	4.21	**0.48**	9.27	3.18	0.46	H
5	6.75	9.88	10.43	2.42	4.74	**0.60**	9.88	3.74	0.52	H
6	4.95	8.95	14.95	1.96	3.02	**0.93**	8.95	2.44	0.90	H
7	10.71	19.15	5.91	9.44	5.71	**1.09**	19.15	19.15	19.15	H
8	4.06	8.66	17.30	1.88	2.66	**1.35**	8.66	2.14	1.29	H
9	4.96	9.14	15.59	2.18	3.12	**1.52**	9.14	2.65	1.50	H
10	2.32	25.03	22.83	**1.53**	2.18	2.16	25.03	26.78	25.01	H
11	1.96	7.67	22.55	**0.78**	2.02	2.02	7.67	1.93	1.91	H
12	2.17	7.20	20.63	**1.07**	1.63	1.63	7.20	1.53	1.51	H
13	1.84	7.69	21.54	**0.63**	1.61	1.58	7.69	22.16	10.80	H
14	4.03	8.71	16.24	1.77	2.75	**1.68**	8.71	2.36	1.67	H
15	7.31	9.88	11.26	2.70	4.09	**0.86**	9.88	3.65	0.83	H

Table 9.8: Patients with cystic fibrosis – rMSE for fitting up to 5% of the initial nitrogen concentration.

no.	exp	explin	pow	exppow	log	loglin	explin	explog	exploglin	H/CF
1	5.59	9.65	11.00	**0.94**	4.78	1.15	9.65	3.83	0.98	CF
2	8.52	10.41	8.64	3.58	4.82	**0.75**	10.41	4.32	0.64	CF
3	7.50	10.06	10.10	2.88	4.39	**0.68**	10.06	3.85	0.59	CF
4	4.18	8.15	9.97	1.22	4.75	**0.57**	8.15	2.93	0.52	2CF
5	7.39	10.14	12.08	3.67	4.29	**0.88**	10.14	3.69	0.88	CF
6	8.29	19.53	6.22	2.67	4.22	**1.17**	19.53	19.53	19.53	CF
7	6.01	9.44	10.36	2.25	5.31	**0.67**	9.44	3.71	0.59	CF
8	3.66	8.66	18.49	2.69	3.05	**1.93**	8.66	2.32	1.79	CF
9	4.84	9.07	14.72	1.57	3.15	**0.93**	9.07	2.54	0.92	CF
10	3.08	8.26	16.64	**0.74**	2.22	1.19	8.26	1.76	1.14	CF
11	2.72	8.17	17.33	**1.20**	2.66	1.48	8.17	1.97	1.38	CF
12	3.73	8.45	15.46	1.89	3.10	**1.32**	8.45	2.22	1.24	CF

Table 9.9: Prediction from 10% to 5% – hypothetical sensors; the intervals are predictions of the terminal breath number by various interval models, len – number of total breaths in file, real – number of real breath end at 5% level, H – healthy person, CF – patient with cystic fibrosis. Prediction intervals $[\underline{a}, \overline{a}]$ containing the true value of breath end having $|\overline{a} - \underline{a}| \leq 2$ are depicted in boldface.

No.	len	real	exp	pow	exppow	loglin	H/CF
1	49	23	[22, 22]	[49, 49]	**[22, 23]**	[18, 19]	H
2	39	23	[20, 20]	[39, 39]	[20, 21]	[17, 18]	H
3	25	14	[12, 12]	[22, 22]	[12, 13]	[11, 12]	H
4	23	13	[11, 11]	[20, 20]	[12, 12]	[11, 23]	H
5	25	14	[11, 11]	[19, 20]	[12, 12]	[12, 25]	H
6	35	21	[17, 18]	[35, 35]	[18, 19]	[16, 17]	H
7	51	51	[21, 21]	**[49, 51]**	[22, 23]	[19, 22]	H
8	32	22	[19, 19]	[32, 32]	[19, 20]	[16, 17]	H
9	32	22	[19, 19]	[32, 32]	[20, 21]	[17, 19]	H
10	51	40	[35, 35]	[51, 51]	[35, 36]	[27, 29]	H
11	51	35	[33, 34]	[51, 51]	**[34, 35]**	[27, 28]	H
12	50	34	[32, 32]	[50, 50]	[32, 33]	[26, 28]	H
13	51	37	**[36, 37]**	[51, 51]	**[37, 38]**	[30, 31]	H
14	36	21	[19, 19]	[36, 36]	**[20, 21]**	[17, 18]	H
15	63	26	[19, 19]	[37, 38]	[21, 22]	[22, 63]	H
1	28	12	[11, 11]	[17, 17]	**[11, 12]**	[28, 28]	CF
2	98	24	[16, 16]	[31, 32]	[17, 18]	[98, 98]	CF
3	80	21	[16, 16]	[30, 30]	[17, 18]	[80, 80]	CF
4	20	8	**[8, 8]**	[12, 12]	**[8, 8]**	[20, 20]	CF
5	48	22	[16, 16]	[31, 32]	[17, 18]	[15, 18]	CF
6	115	61	[37, 37]	[85, 89]	[45, 49]	[115, 115]	CF
7	23	10	[8, 8]	[12, 13]	[9, 9]	[23, 23]	CF
8	32	18	[15, 15]	[32, 32]	[15, 16]	[13, 13]	CF
9	40	19	[16, 17]	[34, 35]	[17, 18]	[15, 17]	CF
10	44	19	[18, 18]	[38, 39]	**[19, 20]**	[16, 18]	CF
11	26	16	[15, 15]	[26, 26]	[15, 15]	[13, 14]	CF
12	31	15	[13, 13]	[24, 25]	[13, 14]	[12, 13]	CF

10 A linear approach to CSP

- Linear relaxation
- Linear programming approach
- Vertex selection for relaxation
- Inner point selection for relaxation
- Properties of the obtained relaxation

In this chapter we introduce one particular approach to solving constraint satisfaction problems over interval boxes. We extend and generalize the work [8] by Araya, Trombettoni and Neveu. We introduce their concept of linear relaxation of a constraint satisfaction problem over a box, which results in a system of real inequalities. The box is then contracted with use of linear programming. To perform the linearization they need to select a vertex point (or a couple of them) of the box. We show that it is possible to select not only vertex points but also any point contained in the contracted box. We show some difficult examples for contractors and consistency techniques, that can be further improved by using the inner point choice. We prove that the proposed linearization is always at least as tight as Jaulin's linearization using two parallel affine functions [97, 99]. The whole chapter is a slightly reworked version of our paper [80]. The aim of this chapter is not to discuss the topic of nonlinear systems at large detail. There are many interesting books and works devoted to this topic, we will mention some of them at the end of this chapter.

10.1 The aim

In this chapter we deal with the *constraint satisfaction problem* (CSP). More specifically, we have a set of equality and inequality constraints

$$f_i(x) = 0, \quad i = 1, \ldots, k, \tag{10.1}$$
$$g_j(x) \leq 0, \quad j = 1, \ldots, l, \tag{10.2}$$

where $f_i, g_j : \mathbb{R}^n \mapsto \mathbb{R}$ are real-valued functions. In compact form, it can be rewritten as

$$f(x) = 0,$$
$$g(x) \leq 0,$$

where $f(x) = (f_1(x), \ldots, f_k(x))$ and $g(x) = (g_1(x), \ldots, g_l(x))$. In global optimization we additionally have a function $\varphi(x)$ and search for the global minimum of the function $\varphi(x)$ subject to these constraints. Such a problem can be transformed to the constraint satisfaction problem (see the next section).

We start with some initial intervals bounding the values of variables $x_1, \ldots, x_n$. The bounding intervals $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n)$ actually form an n-dimensional initial box $\boldsymbol{x}_1 \times \ldots \times \boldsymbol{x}_n$, where we begin the search for solution (or minimum/maximum of $\varphi(x)$).

A common approach is to linearize nonlinear equalities and inequalities first. Such a procedure is called *linear relaxation*. Linear relaxations were also studied in, e.g., [6, 8, 26, 118, 217].

After linear relaxation a system of interval linear inequalities is obtained and linear programming can be used. The result is a box containing the solution that will be hopefully tighter than the initial one. If the box gets tighter, we can iterate this procedure. If the box cannot be tightened, we combine this technique with a branch and bound approach – the current box is split into halves and the procedure is repeated for both parts separately. We can recursively go on with splitting until the size of the box is small enough.

10.2 Global optimization as CSP

The problem of global optimization can be transformed to a constraint satisfaction problem and hence the previously mentioned techniques can be used. Let us have a global optimization problem

$$\begin{aligned} &\min\ \varphi(x), \\ f_1(x) = 0, \ldots, f_k(x) = 0, &\qquad (10.3) \\ g_1(x) \leq 0, \ldots, g_l(x) \leq 0, &\qquad (10.4) \end{aligned}$$

and an initial box $\boldsymbol{x}$. We would like to get a rigorous bounds for $\min \varphi(x)$ for $x \in \boldsymbol{x}$. First by solving the CSP problem defined by (10.3) and (10.4) we get some box $\boldsymbol{x}^*$ where the solution is located. Then we evaluate the $\varphi(x)$ on this box and take the minimum $\underline{\varphi}(\boldsymbol{x}^*)$, this provides a safe lower bound for the global minimum. To obtain an upper bound on the global minimum, we can take any feasible solution x' from $\boldsymbol{x}^*$ and its value $\varphi(x')$. A feasible solution can be found, for example, by local search techniques. As in the previous section, this approach can be combined with a branch and bound approach. That is why in the rest of the chapter we are going to deal with the constraint satisfaction problem only.

10.3 Interval linear programming approach

Our approach is based on linearization of constraints (10.1) and (10.2) by means of interval linear equations and inequalities. Then by using interval linear programming

techniques [68] we construct a polyhedral enclosure to the solution set of (10.1) and (10.2) and contract the initial box $\boldsymbol{x}$. The process can be iterated, resulting in a nested sequence of boxes enclosing the solution set.

Let us have a function $h : \mathbb{R}^n \mapsto \mathbb{R}$ and some interval vector $\boldsymbol{x} \in \mathbb{IR}^n$. Then

$$h(\boldsymbol{x}) = \{h(x) \mid x \in \boldsymbol{x}\}.$$

However, for some more complex functions this is can hardly be computed. We usually compute some enclosure of $h(\boldsymbol{x})$.

First let us choose some point $x^0 \in \boldsymbol{x}$ which will be called a *center of linearization*. Suppose that a vector function $h : \mathbb{R}^n \mapsto \mathbb{R}$ can be enclosed by a linear enclosure

$$h(\boldsymbol{x}) \subseteq S_h(\boldsymbol{x}, x^0)(x - x^0) + h(x^0), \quad \text{for } \forall x \in \boldsymbol{x}, \tag{10.5}$$

for suitable interval-valued function $S_h : \mathbb{IR}^n \times \mathbb{R}^n \mapsto \mathbb{IR}^n$ This is usually calculated by the mean value form as explained in Chapter 3 or [139].

For more efficiency, successive mean value approach ([8]) or slopes ([59, 139]) can be employed. Alternatively, in some situations, a relaxation can be established by analyzing the structure of $h(x)$ – for example, quadratic terms can be relaxed as shown in [118]. After applying such a linearization to all functions $f_1, \ldots, f_k$ and $g_1, \ldots, g_l$ we obtain an interval linear system of equations and inequalities:

$$\begin{aligned} S_f(\boldsymbol{x}, x^0)(x - x^0) + f(x^0) &= 0, & (10.6)\\ S_g(\boldsymbol{x}, x^0)(x - x^0) + g(x^0) &\leq 0. & (10.7) \end{aligned}$$

We can briefly denote it as

$$\begin{aligned} \boldsymbol{A}(x - x^0) &= -f(x^0), & (10.8)\\ \boldsymbol{B}(x - x^0) &\leq -g(x^0). & (10.9) \end{aligned}$$

Theoretically, we do not need to choose the same x^0 for f's and g's. However, we choose the same x^0 for both of them. As the linearization depends on $x^0 \in \boldsymbol{x}$, the question is how to choose x^0.

10.4 Selecting vertices

First, let us take a look at the system (10.8). Using the Oettli–Prager theorem (Theorem 5.4) we can rewrite

$$\boldsymbol{A}(x - x^0) = -f(x^0),$$

as

$$|A^c(x - x^0) + f(x^0)| \leq A^\Delta |x - x^0|.$$

Note that $f(x^0)$ is actually a real number. Now we proceed as in Section 5.2. We can get rid of the first absolute value by rewriting it into the two cases:

$$\begin{aligned} A^c(x - x^0) + f(x^0) &\leq A^\Delta |x - x^0|,\\ -A^c(x - x^0) - f(x^0) &\leq A^\Delta |x - x^0|. \end{aligned}$$

We can get rid of the second absolute value by using knowledge of the sign of each coefficient of the vector in absolute value

$$A^c(x - x^0) + f(x^0) \leq A^\Delta D_{\mathrm{sign}(x-x^0)}(x - x^0),$$
$$-A^c(x - x^0) - f(x^0) \leq A^\Delta D_{\mathrm{sign}(x-x^0)}(x - x^0).$$

Now selection of x^0 should imply knowledge of $\mathrm{sign}(x - x^0)$. The very first idea that can come to our mind is to take x^0 as some corner of the initial box $\boldsymbol{x}$ [8]. If we take, for example, $x^0 = \underline{x}$, we immediately know that $(x - x^0)$ is nonnegative and get the linearization

$$\underline{A}x \leq \underline{A}\underline{x} - f(\underline{x}), \quad \overline{A}x \geq \overline{A}\underline{x} - f(\underline{x}).$$

A similar technique can be applied to the system of inequalities (10.9). We can use the following Gerlach's characterization of all solutions to $\boldsymbol{A}x \leq \boldsymbol{b}$ [53] (cf. [40, 70]).

Theorem 10.1 (Gerlach)**.** *A vector x is a solution of $\boldsymbol{A}x \leq \boldsymbol{b}$ if and only if it satisfies*

$$A^c x - A^\Delta |x| \leq \overline{b}.$$

By applying the theorem to (10.9) we obtain

$$B^c(x - x^0) \leq B^\Delta |x - x^0| - g(x^0).$$

And using the same trick as before we rewrite the absolute value as

$$B^c(x - x^0) \leq B^\Delta D_{\mathrm{sign}(x-x^0)}(x - x^0) - g(x^0).$$

Again, if we set, for example, $x^0 = \underline{x}$, we get the linearization

$$\underline{B}x \leq \underline{B}\underline{x} - g(\underline{x}).$$

The question is, which corner to choose? In [8] it was proved that choosing the corner that gives the tightest linearization is an NP-hard problem. Even if the best corner for linearization was known, it would not guarantee significant contraction gain. However, this gives an insight, how difficult the problem is. Therefore, some heuristics need to be used. According to numerical tests in [8], they propose choosing two opposite corners of $\boldsymbol{x}$ and gathering linear inequalities from both linearizations as the input to a linear program. Which pair of opposite corners is the best choice is an open problem, a random selection seems to do well.

10.5 New possibility: selecting an inner point

Now we are able to linearize according to any corner of the initial box $\boldsymbol{x}$. What about the other points $x^0 \in \boldsymbol{x}$? In the following part we show that also an inner point can be used. Thus we provide an extension of [8].