UNIVERSITY OF TARTU
FACULTY OF SCIENCE AND TECHNOLOGY
INSTITUTE OF MATHEMATICS AND STATISTICS

Reyna María Pérez Tiscareño

# Loss reserving with kernel functions

Financial Mathematics
Master's thesis (15 ECTS)

*Supervisors:*
Meelis Käärik
Liivika Tee

Tartu 2017

**Loss reserving with kernel functions**
Master's thesis
Reyna María Pérez Tiscareño

**Abstract.** Loss reserving is a fundamental concept of actuarial mathematics. A traditionally used method is the chain ladder method. While it is a simple and robust method and works well in many cases, it also has its limitations. The chain ladder method is applied to aggregated data triangle, in a way similar to constructing histograms. Thus, a natural way to improve the approach is to use kernel density estimation instead. This leads to an extension called the continuous chain ladder (CCL) method. In CCL, the main choices a researcher has to make is the choice of the kernel function and the choice of bandwidth, which introduces a suitable level of smoothing. The first choice is usually made for practical or theoretical reasons and it usually has a minor impact on the performance of the estimator. However, the choice of bandwidth can significantly affect the performance of the kernel estimator. There are several possible methods suggested in the literature to choose the bandwidth. To find the optimal bandwidth, the cross-validation procedure is used. A common method for find the optimal bandwidth is the cross-validation. As it is a very time-consuming procedure, some rules of thumb that allow to skip the cross-validation step, can significantly increase the performance of CCL. In this thesis, the main goal is to find the patterns how different input scenarios affect the optimal bandwidths of the CCL model.
**CERCS research specialisation: P160 Statistics, operation research, programming, actuarial mathematics**
**Keywords:** Chain ladder method, Claim reserving, Continuous Chain ladder, Kernel smoothing.

**Kahjureservide hindamine kasutades tuumafunktsioone**
Magistritöö
Reyna María Pérez Tiscareño

**Lühikokkuvõte.** Reservide hindamine on üks kindlustusmatemaatika põhiülesandeid. Laialdaselt kasutatatud klassikaline ahel-redel meetod on lihtne ja robustne, kuid tal on omad piirangud. Arvestades, kuidas ahel-redel meetodit summeeritud arengukolmnurgale rakendatakse, võib seda käsitleda kui teatavat histrogrammi-tüüpi lähendamiseülesannet. Loogiline järgmine samm on histogrammipõhisel lähenemisel kasutada tiheduse hindamist tuumafunktsioonide abil. Sedasi jõuame pideva ahel-redel meetodini. Tiheduse hindamise juures on olulised küsimused tuuma valik ja aknalaiuse määramine. Seejuures tuuma enda mõju on reeglina väike võrreldes aknalaiuse mõjuga. Optimaalse aknalaiuse määramiseks kasutatakse enamasti ristvalideerimist. Teadaolevalt on see väga ressursimahukas meetod, mistõttu teatud rusikareeglite leidmine, mis võimaldaksid ristvalideerimise sammu vahele jätta, tõstaks oluliselt pideva ahel-redel meetodi rakendamise kiirust. Käesoleva töö põhieesmärk ongi leida mustreid, kuidas erinevad tingimused mõjutavad pideva ahel-redel mudeli optimaalset aknalaiust.
**CERCS teaduseriala: P160 Statistika, operatsioonanalüüs, programmeerimine, finants- ja kindlustusmatemaatika.**
**Märksõnad: ahel-redel meetod, kahjureservide hindamine, pidev ahel-redel, tuumaga silumine.**

# *Acknowledgements*

I would like to thank God and all the persons that teach me something every day. Thanks to my advisors, Meelis Käärik and Liivika Tee who introduced me this topic and helped me to understand the ideas. Also I would like to thank the teachers of the master studies for transmitting us their knowledge. Also, I thank Mart Abel for everything. Thanks to my parents, my brothers and my sister who have shown me that with effort I can reach all my goals. I would like also to thank Mati Abel and Elts Abel for all their help.

# Contents

# Introduction

For insurance companies it is very important to estimate accurately loss reserves, they need to be able to fulfill their contractual obligations to policyholders. Traditionally actuaries have worked in this estimation with the Chain Ladder Method (CLM), in general this method works well in many cases but it has to be adjusted manually in many other cases. A key feature of the majority of claims reserving methods used in practice, including the CLM, is that they assume that the data have been aggregated. This gives that the models are easier to handle but it can add some weakness for that reason several researchers have tried to work with detailed micro-level information.

As the chain ladder method is a most widely used loss reserving model among practitioners, then it is not a surprise that the CLM has been extended and combined with other methods.

In this thesis we will apply a method that uses data recorded in continuous time but based on the CLM philosophy, for that reason it is called "Continuous Chain Ladder method". This method was introduced in [3], where the authors show that the classical CLM can be regarded as a structured histogram on a triangle. The original CLM groups the data and develops a multiplicative histogram model but the continuous chain ladder method does not group the data, so it can be treated as having a density on the triangle.

So, the histogram type of estimator can be improved in the same way that mathematical statisticians normally would improve histogram estimators: by introducing smoothing. To introduce such smoothing, in this thesis, we will consider the kernel density estimator.

The kernel estimation of probability density functions depends on the kernel $K$ and the bandwidth $h$. The kernel determines the shape of the weighting function and the bandwidth determines the "width" of the weighting function and hence the smoothing. Kernel density estimators are sensitive to the choice of bandwidth, but the choice of a kernel function does not usually affect the results considerably. Thus, in this thesis we investigate how the selection of bandwidths is affected by different combinations of input parameters.

In this thesis we will work only with simulations of incurred but not reported claims. First, the data is simulated on monthly basis. We will generate the complete rectangle and estimate the reserve (it could be considered as the reserve calculated using the real future data). Then we consider only the upper triangle and using the continuous chain ladder method and the kernel density estimator, we predict the lower triangle. Next, we calculate the reserve estimate. Finally we compare values of both, estimated reserve and actual reserve.

The thesis is structured as follows: to understand the methods that we will use, the Chapter 1 introduces to the reader the reserving problem and the classical chain ladder method. In Chapter 2 we explain the kernel density estimation ideas. Chapter 3 clarifies the ideas of the continuous chain ladder method which is a quite recent method. In Chapter 4, the simulation study is carried out and the choice of the best bandwiths in all fixed combinations of input parameters is done. Finally, Chapter 5 gives the conclusion of the simulation study developed in Chapter 4.

# Chapter 1

# The reserving problem and the classical chain ladder method

## 1.1 Claims reserving

An insurance company needs to have a reserve, this is an amount set aside to meet company's principal insurance liabilities. Reserves have several purposes, for example, they enable the company to meet and administer its contractual obligations to policyholders, also they can be used in order to set future premium rates.

After the claim event has occurred (accident date $T_1$, must be within the insurance contract period), the policyholder will report the incident to the insurance company (reporting date $T_2$, $T_2 \geqslant T_1$). In due course the insurer will make any payments required. There may be several payments made under a single claim. When the insurer considers that there will not be more payments for this claim, the claim file is closed (claims closing date $T_3$, $T_3 \geqslant T_2$).

Let $t$ be the today's time point. We will say that when $T_1 \leqslant t < T_2$, the claim is incurred but not yet reported to the insurer (**IBNR** claim) and we will say that the claim is reported but not settled, when $T_2 \leqslant t < T_3$ (**RBNS** claim), when $T_3 \leqslant t$, we say that the claim is settled and no further development is expected.

To know how much to put aside in a reserve, it is needed to estimate the size and the frequency of the future claims but also something that could be taken in account is that: non-life insurance claims cannot be settled immediately. There could be a reporting delay (days, weeks, months or even years) or/and settlement delay. So, it makes harder for an insurance company to try to estimate the exact figure for total claims each year with as much confidence and accuracy as possible.
These claim reserves should meet the following requirements:

1) They should be sufficiently high, so that all liabilities can be fulfilled.

2) They should not be too high such that shareholders get an appropiate dividend.

3) They should be best-estimate such that they can be used for the pricing of future insurance contracts.

The regulator should supervise and review these requirements regularly (to protect policyholders and/or insured).

Accurate loss reserves are essential for insurance companies to meet and administer their contractual obligations to policyholders. The reserves are related to claims at different stages in the settlement process, in this work we will deal with reserves required for IBNR claims.

There are several ways for presenting claims data. Here, the claims will be presented as a triangle (run-off triangles), which is the most common used method. The claim development pattern is defined by the choice for the origin period which can be given by important dates of the claims' life cycle:

- underwriting;
- claim occurrence;
- reporting claim;

and by the development period, which can be

- claim occurrence;
- reporting claim;
- claim settlement (year of final payment).

Depending on the choice of origin period and development period, the run-off triangles can describe the development of different stages of different reserves.

If the development period is the reporting claim period, then the corresponding run-off triangle describes development in IBNR reserve, but if the development period is the claim settlement period, then the corresponding run-off triangle describes development in RBNS reserve.

Usually, the information is provided in an aggregated way where, in theory, any aggregation periods could be considered, such as quarters, months, years etc. As we explained before, depending on the data being considered, each cell in the triangle could contain the number of reported or paid claims or aggregated payments (see Table 1.1).

In this thesis we will consider months, the origin period will be the claim occurrence and the development period the number of months that passed from the claim occurrence until the reporting claim moment.

The basic chain ladder method is a method for projecting run-off triangles, this traditional method is the most widely used loss reserving model. We will talk about it in the next section.

Run-off triangles are important in the practical work of actuaries who want to forecast future claim numbers and amounts.

## 1.2 Chain Ladder Method

The so called reserve estimating outstanding liabilities is the single most important number in a non-life insurance balance sheet. In order to introduce the basic chain ladder method, we will use the following notation:

- **accident years** are denoted by $i \in \{1, \dots, I\}$;
- **development years** are denoted by $j \in \{0, \dots, J\}$ with final settlement delay $J$. We will consider that $J = I - 1$;
- **incremental claims** are denoted by $X_{i,j}$;
- **cumulative claims** are denoted by

$$C_{i,j} = \sum_{l=0}^{j} X_{i,l};$$

- **ultimate (nominal) claims** are denoted by $C_{i,J}$ (last column).

TABLE 1.1: Run-off triangle

| **Accident year i/ Development year j** | **0** | **1** | **2** | **...** | | **j** | **...** | **J** |
|---|---|---|---|---|---|---|---|---|
| **1** | | | | | | | | |
| **2** | | | | | | | | |
| ⋮ | | | | | | | | |
| **i** | | | | | | | | |
| ⋮ | | | | | | | | |
| **I-1** | | | | | | | | |
| **I** | | | | | | | | |

- **upper triangle** $\mathcal{D}_I = \{C_{i,j} : i + j \leqslant I\}$; (observations in the past).
- **lower triangle** $\mathcal{D}_I^c = \{C_{i,j} : i + j > I\}$; (observations to be predicted).
- **diagonals** $k = i + j$ are the accounting years with $k = 1, \dots, I + J$.

Now, we are ready to explain the idea of the chain ladder method. Assume that for all the development years $j = 0, \dots, J - 1$ there exist $f_j > 0$ such that

$$C_{i,j+1} \approx f_j C_{i,j} \text{ for all accident years } i = 1, \dots, I.$$

The factor $f_j$ is called *chain ladder method (CLM) factor* (it is also known by *age-to-age factor, link ratio*). So, individual link ratios are given by

$$C_{i,j+1}/C_{i,j} \approx f_j.$$

We would like to determine the CLM factors $f_j$ for all $j = 0, \dots, J-1$. We set

$$\hat{f}_j^{CL} = \frac{\sum_{i=1}^{I-j-1} C_{i,j+1}}{\sum_{i=1}^{I-j-1} C_{i,j}}$$

for the CLM factor estimator $\hat{f}_j^{CL}$ of $f_j$, given observations $\mathcal{D}_l$. Now, we will predict the lower triangle $i + j > I$ by

$$\hat{C}_{i,j}^{CL} = C_{i,I-i} \prod_{l=I-i}^{j-1} \hat{f}_l^{CL},$$

where $C_{i,I-i}$ is the last observed diagonal. Finally, CLM claims reserves for accident years $i = I - J + 1, \dots, I$ are defined by

$$\hat{R}_i^{CL} = \hat{C}_{i,j}^{CL} - C_{i,I-i}.$$

The classical CLM can be seen as a histogram on a triangle. This method groups the data and develops a multiplicative histogram model. This idea will be more thoroughly explained later.

The histogram separates the data into distinct non-overlapping bins and constructs bars (hypercubes) with heights defined as the proportion (or number) of observations falling into each bin. This proportion gives an estimate of the probability density function in the mid point of the bin.

We will briefly describe the idea of the histogram associated to a two-dimensional scenario (this was introduced in [3]). Let $Z_1, Z_2, \cdots, Z_n$ be an i.i.d. (independent and identically distributed) random sample for a population $Z = (X, Y)^t$, having bivariate continuous density $f$. The support of $f$ (in the CLM, the support is a triangle) is divided into $n$ squares with length of the sides $\Lambda$.

Thus, the histogram estimator at any point $z_0 = (x_0, y_0)^t$ in the support of $f$ is defined by

$$\hat{f}(z_0) = \nu(z_0)/n\Lambda^2, \tag{1.2.1}$$

where $\nu(z_0)$ is the number of sample data falling in the square which contains $z_0$.

Since the chain ladder method approach can be seen as a histogram type of estimator, then we can improve that estimator by introducing smoothing. In the next chapter we will talk about the kernel estimator which is a way to improve histogram estimator. We will also describe a kernel density estimator for the histogram introduced above (see Chapter 2, 2.2.2).

# Chapter 2

# Kernel Density Estimation

## 2.1 Density estimation and histograms

In this section we will talk about some methods for estimating the probability density function (pdf) from a sample.

The probability distribution of a continuous-valued random variable $X$ is described in terms of its pdf, $f(x)$. An objective of many investigations is to estimate $f(x)$ from a sample of observations $x_1, \cdots, x_n$. We will assume that the observations can be regarded as independent realization of $X$.

One method to get $f(x)$ is the histogram. This method reminds a density function divided into classes, it does not approximate a density function but it can be rescaled to make it to be an estimate of the density. The histogram is the basis of kernel density estimation.

### 2.1.1 Some approaches to the estimation of the pdf

The idea of the *parametric approach* for estimating $f(x)$ is to assume that $f(x)$ is a member of some parametric family of distribution, for example, $N(\mu, \sigma^2)$, and then to estimate the parameters of the assumed distribution using the data. This approach has advantages if the distributional assumption is correct or if at least it is not seriously wrong.

The main disadvantage of the parametric approach is lack of flexibility. Each parametric family of distributions imposes restrictions on the shapes that $f(x)$ can have.

The idea of the *non-parametric approach* is to avoid restrictive assumptions about the form of $f(x)$ and to estimate this directly from the data.

A well-known non-parametric estimator of the pdf is the histogram. It has the advantage of simplicity but is also has disadvantages, such as lack of continuity. Moreover, in terms of various mathematical measures of accuracy, there exist alternative non-parametric estimators that are superior to histograms.

The sensitivity of the histogram to the placement of the bin edges is a problem not shared by other density estimators such as the kernel density

estimator which will be introduced in this chapter. The bin edge problem is one of the histogram's main disadvantages.

One method to get $f(x)$ by giving the sample is described next. Find the sample minimum and maximum, then to divide the interval between them into subintervals (right-closed and left-open) with equal length (but could be considered also with different lengths) and count for each interval how many sample values are in the interval. We represent these frequencies by bars.

To understand the idea better, we consider the following example, where the data set was taken from the predetermined data in R "Motor Trend Car Road Tests". This data was extracted from the 1974 Motor Trend US magazine and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973–74 models). From this data we consider the cars weight (1000 lbs), this is mtcars$wt,

$$2.620\ \ 2.875\ \ 2.320\ \ 3.215\ \ 3.440\ \ 3.460\ \ 3.570\ \ 3.190\ \ 3.150\ \ 3.440\ \ 3.440\ \ 4.070$$

$$3.730\ \ 3.780\ \ 5.250\ \ 5.424\ \ 5.345\ \ 2.200\ \ 1.615\ \ 1.835\ \ 2.465\ \ 3.520\ \ 3.435\ \ 3.840$$

$$3.845\ \ 1.935\ \ 2.140\ \ 1.513\ \ 3.170\ \ 2.770\ \ 3.570\ \ 2.780.$$

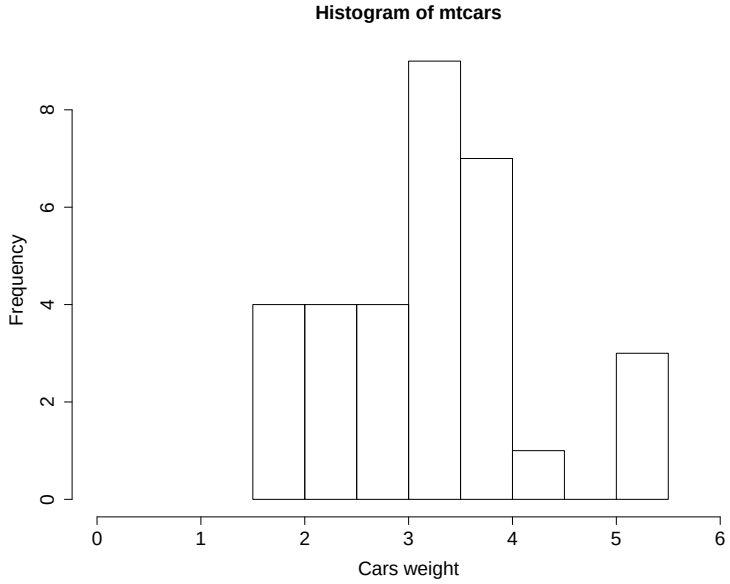



FIGURE 2.1: Histogram

In the Figure 2.1 we see the histogram for the example considering subintervals with equal length.

If the sample size and the number of intervals are enough large then this graph (histogram) reminds us a density function which is divided into classes. We observe that such graph does not approximate a density function because the area under the graph is not equal to one but we can rescale the histogram to make it to estimate the density (see Figure 2.2).

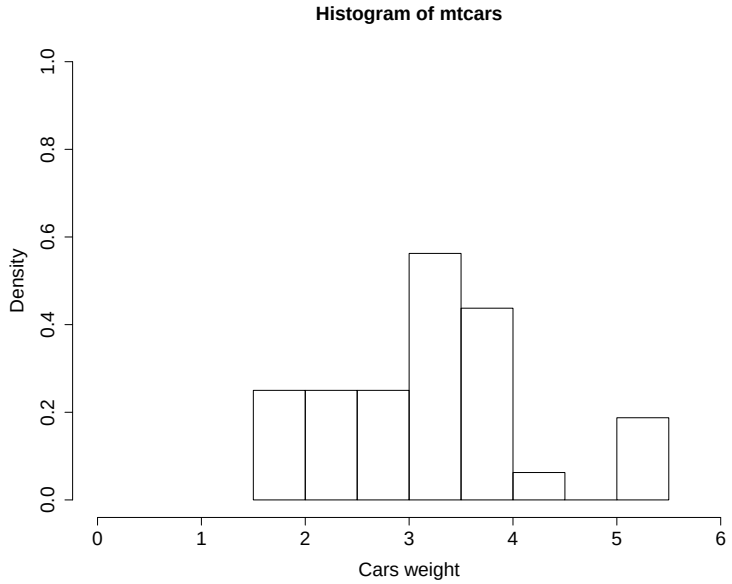



FIGURE 2.2: Density estimate

Let $x_1, x_2, \ldots, x_n$ be empirical distribution sample points with probabilities

$$p(x_1) = p(x_2) = \ldots = p(x_n) = \frac{1}{n}.$$

In order to transform ("spread out") the probabilities by histogram, we fix a subinterval with length $w$ such that it contains $k$ sample points. Then we spread the probability $\frac{k}{n}$ over this subinterval. Thus, the density estimate for that subinterval is $\frac{k}{nw}$ and does not depend on how the sample points are located in the subinterval. This "spreading out" is the basis of kernel density estimation.

We notice that it is not interesting to draw a histogram with a single observation per class. Therefore, it may be useful to draw a histogram where the subintervals have different length. Namely, we divide the interval between sample minimum and sample maximum in $k$ subintervals with lenghts $w_1, \ldots, w_k$, respectively. Let $p_1, \ldots, p_k$ be the probabilities over the subintervals. Then the graph of bars with respective heights $\frac{p_1}{w_1}, \ldots, \frac{p_k}{w_k}$ gives us an estimate for the density (the area under the graph is one).

Following, we summarize the ideas discussed before to get the estimator of $f(x)$. To have a histogram, one needs to select a left bound, or starting point, $x_1$, and the bin width, $h$, usually called the *binwidth*. The bins are given by $[x_1 + (i-1)h, x_1 + ih)\,,\ \ i = 1, 2, \ldots, n$. The estimator for $f(x)$ is given by

$$\hat{f}(x) = \frac{1}{n}\ \frac{\text{Number of observations in the bin where } \ x \ \text{ is contained}}{h}.$$

More generally, one can also use bins of different widths, in which case

$$\hat{f}(x) = \frac{1}{n} \frac{\text{Number of observations in the bin where } x \text{ is contained}}{\text{Width of bin that contains } x}.$$

Two choices have to be made when constructing a histogram; the binwidths and the positioning of the bin edges. These choices can have a substantial effect on the shape and other properties of $\hat{f}(x)$.

The binwidth $h$ is usually called a *smoothing parameter*, since it controls the amount of "smoothing" being applied to the data. We will see that for kernel estimators, the scale of the kernel plays a role analogous to that of the binwidth.

## 2.2 Weight functions

By definition of pdf, $f(x)$, of a random variable, $X$, we have that

$$P(x-h < X < x+h) = \int_{x-h}^{x+h} f(t)dt \approx 2hf(x).$$

Hence,

$$f(x) \approx \frac{1}{2h} P(x-h < X < x+h).$$

So, if we estimate the probability by a relative frequency in the sample, then

$$\hat{f}(x) = \frac{1}{2h} \frac{\text{number of observations in } (x-h, x+h)}{n}$$

and $\hat{f}(x)$ can be written also as follows:

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^{n} w(x - x_i, h), \tag{2.2.1}$$

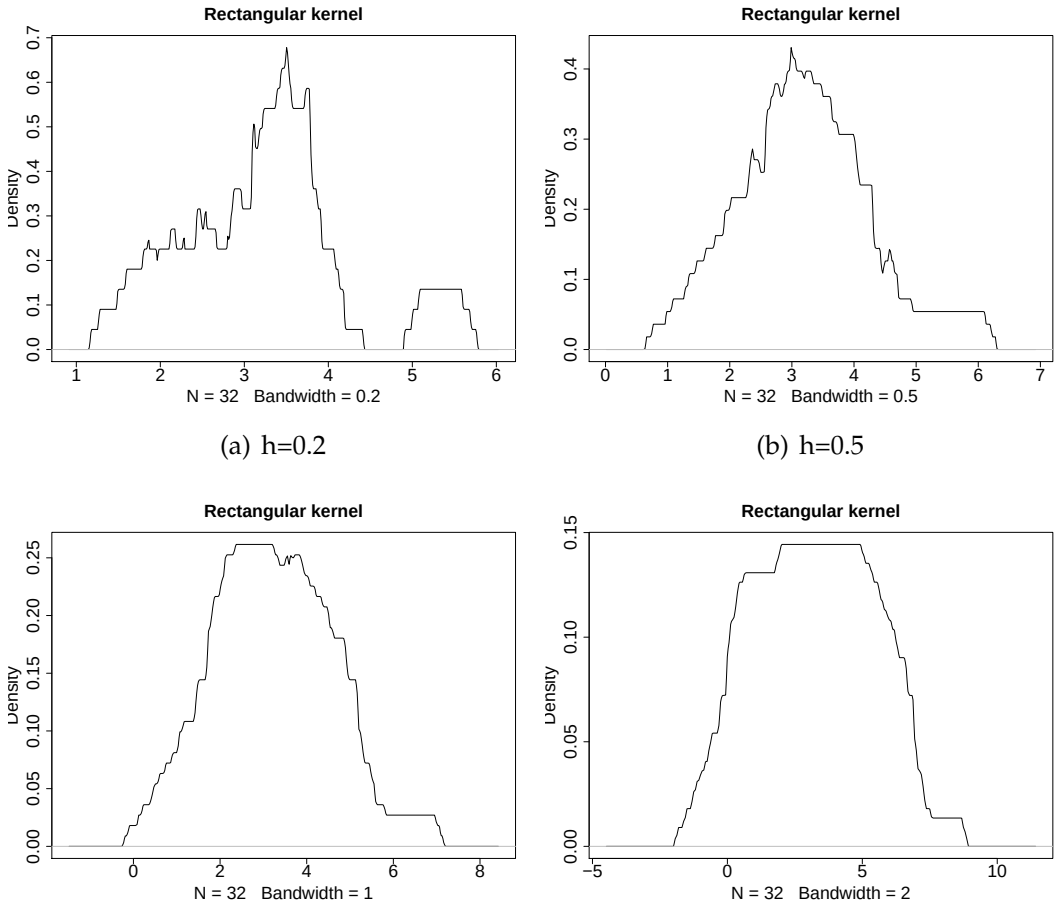
where $x_1, x_2, , \ldots x_n$ are the observed values and

$$w(t,h) = \begin{cases} \frac{1}{2h} & \text{for} \quad |t| < h, \\ 0 & \text{otherwise.} \end{cases}$$

To give a better idea about 2.2.1, we can imagine that the center of the base of a rectangle (with height $\frac{1}{2h}$ and width $2h$) is placed over each observed point on the $x$-axis.

The estimate of the pdf at a given point is $\frac{1}{n}$ times the sum of the heights of all the rectangles that cover the point.

The Figure 2.3 shows $\hat{f}(x)$ for rectangular "weighting function", for different values of $h$.




(a) h=0.2 (b) h=0.5

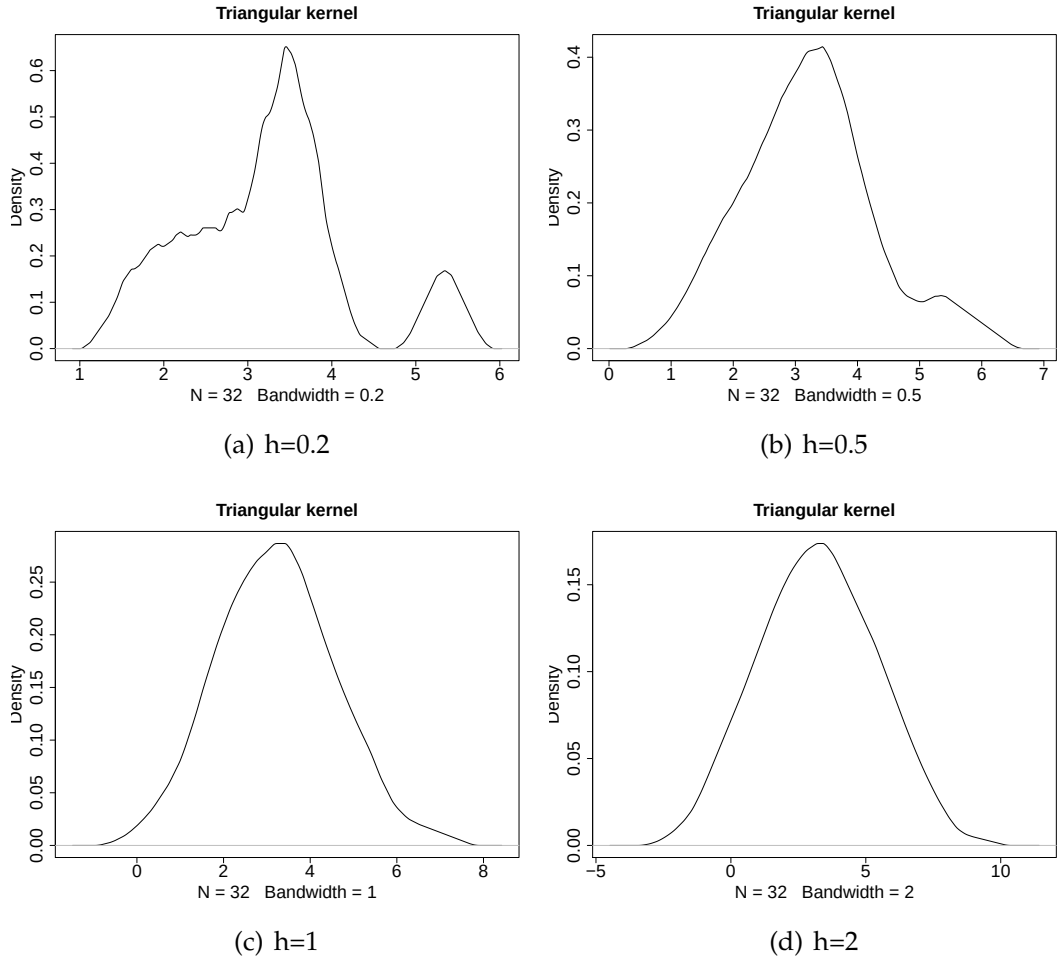
(c) h=1 (d) h=2

FIGURE 2.3: Rectangular kernel

Instead of using rectangles in 2.2.1, we can use other weighting functions, for example triangles:

$$w(t,h) = \begin{cases} \frac{1}{h}\left(1 - |t|/h\right) & \text{for} \quad |t| < h, \\ 0 & \text{otherwise.} \end{cases}$$

The Figure 2.4 shows $\hat{f}(x)$ for triangular "weighting function" for different values of $h$.




(a) h=0.2 (b) h=0.5

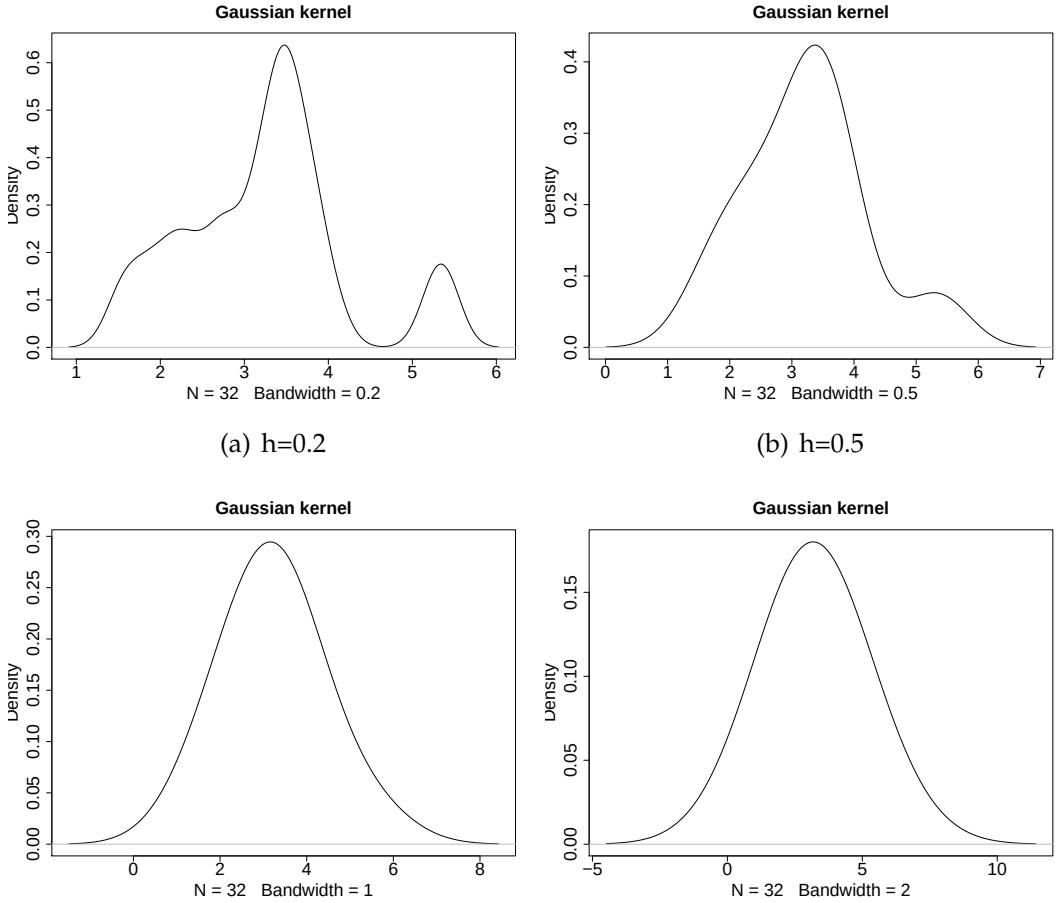
(c) h=1 (d) h=2

FIGURE 2.4: Triangular kernel

Another alternative weighting function is the Gaussian:

$$w(t,h) = \frac{1}{\sqrt{2\pi}h} e^{-t^2/2h^2}, \qquad -\infty < t < \infty.$$

The Figure 2.5 shows $\hat{f}(x)$ for Gaussian "weighting function" for different values of $h$.




(a) h=0.2 (b) h=0.5

(c) h=1 (d) h=2

FIGURE 2.5: Gaussian kernel

We observe that the fluctuation in $\hat{f}(x)$ decreases with increasing of $h$.

The weighting functions, $w(t,h)$, defined before, have the form

$$w(t,h) = \frac{1}{h} K\left(\frac{t}{h}\right),$$

where $K$ is a function of a single variable called the *kernel*.

We observe that the kernel is the weighting function with $h = 1$ and it determines the shape of the weighting function. The parameter $h$ is called the *bandwidth* or *smoothing constant*.

As examples of kernel functions, we have the rectangular, triangular and Gaussian kernels. Defined when $h = 1$, in the definitions of rectangular, triangular and Gaussian weighting functions, respectively.

A *kernel density estimate* is defined by

$$\hat{f}(x) = \frac{1}{h}\sum_{j=1}^{n} p(x_j) K\left(\frac{x - x_j}{h}\right),$$

where $K(x)$ is a kernel function. We can see that kernel estimation of pdfs depends of the kernel, $K$, and the bandwidth, $h$. The kernel determines the shape of the weighting function and the bandwidth determines the "width" of the weighting function and hence the smoothing. The properties of $\hat{f}(x)$ are determined by $K$ and $h$, this is the reason why several researchers study how to select $K$ and $h$ in order to optimize the properties of $\hat{f}(x)$. Kernel density estimators are sensitive to the choice of bandwidth, but the choice of a kernel function does not usually affect the results considerably.

For the histogram described in Chapter 1 (see Chapter 1, 1.2.1), the kernel estimator for any estimation point $z_0 = (x_0, y_0)$ in the support, is defined as

$$\hat{f}_h(z_0) = n^{-1}\sum_{i=1}^{n} K_h(z_0 - Z_i), \tag{2.2.2}$$

where $K_h(\cdot)$ is a two-dimensional kernel and $h = (h_1, h_2) \in \mathbb{R}^2_+$ is a bandwith parameter, where $h_1$ controls the origin direction and $h_2$ controls the development direction. In this work we will consider

$$K_h(x, y) = K_{h_1}(x) K_{h_2}(y),$$

where $K_{h_1}(x) = h_1^{-1} K(x/h_1)$, $K_{h_2}(y) = h_2^{-1} K(y/h_2)$ and $K$ is an one-dimensional kernel function.

## 2.3 Properties of kernel estimators

There are different ways to quantify the accuracy of a density estimator. In this work we will focus on the mean squared error (MSE) and its two components (bias and standard error (or variance)).

$$MSE(\hat{f}(x)) = E(\hat{f}(x) - f(x))^2 = (E\hat{f}(x) - f(x))^2 + E(\hat{f}(x) - E\hat{f}(x))^2$$

$$= Bias^2(\hat{f}(x)) + Var(\hat{f}(x)).$$

A measure of the global accuracy of $\hat{f}(x)$ is the mean integrated squared error (MISE)

$$MISE(\hat{f}) = E\int_{-\infty}^{\infty}(\hat{f}(x) - f(x))^2 dx.$$

Considering a kernel density estimate $\hat{f}(x)$ with bandwidth $h$, it is shown (see [7]) that

$$Bias(\hat{f}(x)) \approx \frac{h^2}{2}k_2 f''(x),$$

where $k_2 := \int_{-\infty}^{\infty} z^2 K(z)dz$ and

$$Var(\hat{f}(z)) \approx \frac{1}{nh}f(x)\int_{-\infty}^{\infty} K^2(z)dz.$$

Now

$$MSE(\hat{f}(x)) = Bias^2(\hat{f}(x)) + Var(\hat{f}(x)) \approx \frac{1}{4}h^4 k_2^2 f''(x)^2 + \frac{1}{nh}f(x)j_2, \quad (2.3.1)$$

where $j_2 := \int_{-\infty}^{\infty} K^2(z)dz$. Integrating 2.3.1, we get that

$$MISE(\hat{f}(x)) \approx \frac{1}{4}h^4 k_2^2 \beta(f) + \frac{1}{nh}j_2, \quad (2.3.2)$$

where $\beta(f) := \int_{-\infty}^{\infty} f''(x)^2 dx$.

We observe that $MISE(\hat{f})$ changes as a function of the bandwidth $h$. The optimal value of $h$, which minimizes $MISE(\hat{f})$, is:

$$h_{opt} = \left(\frac{\gamma(K)}{n\beta(f)}\right)^{\frac{1}{5}}, \quad (2.3.3)$$

where, $\gamma(K) := j_2 k_2^{-2}$. Substituting $h$ by $h_{opt}$ in 2.3.2, we get that

$$MISE_{opt}(\hat{f}) = \frac{5}{4}\left(\frac{\beta(f)j_2^4 k_2^2}{n^4}\right)^{\frac{1}{5}}. \quad (2.3.4)$$

The $MISE(\hat{f})$ can also be minimized with respect to the kernel used. In [5] it is shown that the Epanechnikov kernel is the most efficient kernel in this respect. The Epanechnikov kernel is defined as the function

$$K(u) = \begin{cases} \frac{3}{4}(1-u^2) & \text{for} \quad |u| \leqslant 1, \\ 0 & \text{otherwise.} \end{cases}$$

By this result and 2.3.4, we can examine the impact of the kernel choice on

$MISE_{opt}(\hat{f})$. The efficiency of a kernel $K$, relative to the optimal Epanechnikov kernel, which will be denoted by $K_{EP}$, is defined as the quotient

$$\left( \frac{MISE_{opt}(\hat{f}) \text{ using } K_{EP}}{MISE_{opt}(\hat{f}) \text{ using } K} \right)^{\frac{5}{4}}.$$

### 2.3.1 Selection of the bandwidth

The selection of the bandwidth of the kernel estimator is a subject of considerable research. There are several methods to do it. We will enounce some of them:

- Subjective selection: one can experiment by using different bandwidths and simply select one that "looks right" for the type of data under investigation.
- Selection with reference to some given distribution: one selects the bandwidth that would be optimal for a particular pdf.
- Cross-validation: by definition,

$$MISE(\hat{f}(x;h)) = E \int (\hat{f}(x;h) - f(x))^2 dx$$

$$= E \int \hat{f}(x;h)^2 dx - 2E \int \hat{f}(x;h) f(x) dx + \int f(x)^2 dx.$$

As the third term does not depend on the bandwith, then to minimize $MISE(\hat{f}(x;h))$ is equivalent to minimize

$$MISE(\hat{f}(x;h)) - \int f(x)^2 dx = E\left[ \int \hat{f}(x;h)^2 dx - 2 \int \hat{f}(x;h) f(x) dx \right].$$

An approximately unbiased estimator of the right side is given by

$$\hat{LSCV}(h) = \int \hat{f}(x;h)^2 dx - \frac{2}{n} \sum_{i=1}^{n} \hat{f}_{-i}(x_i;h),$$

where

$$\hat{f}_{-i}(x;h) = (n-1)^{-1} \sum_{j \neq i}^{n} K_h(x - x_j),$$

this is the estimated density at the argument $x$ using the original sample apart from observation $x_i$.

We choose, $h_{opt}$, which minimize $\hat{LSCV}(h)$.

- "Plug-in" estimator: the idea is to estimate the bandwith $h$ from 2.3.3 by applying a separate smoothing technique to estimate $f''(x)$ and hence $\beta(f)$.

# Chapter 3

# The continuous chain ladder method

The majority of claims reserving methods used in practice, including the CLM, assume that the data has been aggregated. This aggregation is often done by year, but it could be done by quarter or month. We note that this aggregation implies some pre-smoothing of the data.

The continuous chain ladder method (CCL) was defined in [3] and uses data recorded in continuous time. The philosophy underlying the CLM, it can be viewed as a continuous version of the CLM. We showed in the Chapter 1 that the CLM groups the data and we can give a multiplicative histogram model. This idea will be simplified following the same approach without grouping the data. When the data is not grouped, it can be treated as having a density on the triangle. This multivariate density can be estimated using local linear smoothing methods and it can be approximated on the triangle by a multiplicative density.

The continuous chain-ladder is the first chain-ladder extension up to our knowledge that does not assume the data aggregation and the use of continuous time. As it was mentioned, this model assumes that the data is recorded in continuous time but in practice this kind of data collection can not be assumed from an insurance company. The lowest level of data aggregation that could be considered, is daily data. In [2] is investigated how big is the gain of a micro-level approach and how much do the different levels of data aggregation influence the reserve estimates.

## 3.1 Continuos chain ladder method

We will introduce the general ideas of the continuous chain ladder method (CCL). In the CLM, the data is given in a triangle and with some aggregation, in CCL the data is considered continuous and without aggregation. So the data is given as a two dimensional space, but it still forms a triangle. Moreover, the data can be treated as having a density on the triangle, namely, we consider a random sample of $n$ i.i.d. (independent and identically distributed random variables) observations $\{(X_i, Y_i) \mid i = 0, \ldots, I; j = 0, \ldots, I-1\}$ from

a population $(X, Y)$ having a density $f$ with support on a subset

$$\mathcal{L} = \{(x, y) \mid 0 \leqslant x, 0 \leqslant y, x + y \leqslant I\}$$

of the square $\mathcal{S} = \{(x, y) \mid 0 \leqslant x, y \leqslant I\}$, where $x$ is the underwriting (or accident) time and $y$ is the claims development time within the time frame $[0, I]$, $I > 0$, we considered that the origin period starts from zero by simplicity. By [3], reserving can be seen as a density estimation problem where the claims triangle defines the boundary region. As the kernel methods have boundary problem, then the triangular support requires refined boundary corrections methods to be considered (the support of variables is bounded and the standard kernel estimator continues to give weight outside the supports). Some researchers have worked in the solution to this problem but it was chosen in the thesis the local linear estimator introduced in Nielsen (1999,[4]) which proposes a non-parametric multivariate density estimation with arbitrary boundary regions.

The Nielsen's local linear estimator is obtained at each point $(x_0, y_0) \in \mathcal{L}$ by solving the minimization problem:

$$\hat{\Theta} = \arg \min_{\Theta} \lim_{b \to 0} \int_{\mathcal{L}} \{\tilde{f}_h(z) - \Theta_0 - \Theta_{1,1}(x - x_0) - \Theta_{1,2}(y - y_0)\}^2 K_{h_1}(x - x_0) K_{h_2}(y - y_0) dz, \tag{3.1.1}$$

where $\Theta = (\Theta_0, \Theta_{1,1}, \Theta_{1,2})$, $\hat{\Theta} = (\hat{\Theta}_0, \hat{\Theta}_{1,1}, \hat{\Theta}_{1,2})$ and

$$\tilde{f}_h(z) = n^{-1} \sum_{i=1}^{n} K_{h_1}(X_i - x_0) K_{h_2}(Y_i - y_0)$$

is the standard kernel estimator at the point $z = (x, y) \in \mathcal{L}$, with bandwidth parameters $h = (h_1, h_2) \in \mathbb{R}^2_+$ (see Chapter 2, 2.2.2). The local linear density of $f(x, y)$ is given by

$$\hat{f}(x, y; h) = \hat{\Theta}_0, \quad \text{where} \quad h = (h_1, h_2) \in \mathbb{R}^2_+.$$

We observe that this estimator is defined on $\mathcal{L}$, but we would like to estimate the density function on $\mathcal{S}$. In order to get it, we assume a multiplicative structure for the density. Namely, the density $f(x, y) = f_1(x) f_2(y)$, where $(x, y) \in \mathcal{S}$. Next, we will provide an estimator for the triangle $\mathcal{S} - \mathcal{L}$ though a two-step method.

a) From the available data estimate the estimator $\hat{f}_h$ in the set $\mathcal{L}$ (from the minimization problem described above).

b) Assume that $f(x, y) = f_1(x) f_2(y)$, where $(x, y) \in \mathcal{S}$ and estimate $f_1$ and $f_2$ by solving the minimization:

$$min_{f_1, f_2} \int_{\mathcal{L}} (\hat{f}_h(x, y) - f_1(x) f_2(y))^2 w(x, y) dx dy,$$

where $w(x, y)$ is a weighting function.

Now, we will describe how to solve the minimization problem described in $b)$, we consider that in $b)$, $w(x, y) = (\hat{f}_h(x, y))^{-1}$, then the minimization problem is solved using the following recursive algorithm:

1) We denote by $\hat{f}^{(0)}$ the estimator found in $a)$ and we will consider the initial estimator of $f_1$, which will be denoted by $\hat{f}_1^{(0)}$.

2) As, $f(x, y) \approx \hat{f}_1^{(0)}(x) f_2(y)$ then

$$\int_{\mathcal{J}_y} f(x, y)dx \approx f_2(y) \int_{\mathcal{J}_y} \hat{f}_1^{(0)}(x)dx,$$

where $\mathcal{J}_y = \{x|(x, y) \in \mathcal{J}\}$. Hence, we can estimate the density $f_2$ by

$$\hat{f}_2^{(1)}(y) = \frac{\int_{\mathcal{J}_y} \hat{f}^{(0)}(x, y)dx}{\int_{\mathcal{J}_y} \hat{f}_1^{(0)}(x)dx}.$$

3) We calculate the updated estimator for $f_1$ by

$$\hat{f}_1^{(1)}(x) = \frac{\int_{\mathcal{J}_x} \hat{f}^{(0)}(x, y)dx}{\int_{\mathcal{J}_x} \hat{f}_2^{(1)}(y)dy},$$

where $\mathcal{J}_x = \{y|(x, y) \in \mathcal{J}\}$.

We have to repeat 2)-3) until the desired convergence criterion is achieved. This provides estimates for any point in $\mathcal{S}$.

The CCL method is developed for individual (or continuous) claims data but it can be still used to aggregated data. Following we will describe how could be applied the CCL to aggregated data.

The aggregated cumulative incurred claims triangle can be represented by

$$\{C_{i,j} \mid (i, j) \in \nabla\},$$

where $\nabla = \{(i, j) \mid i = 1, \cdots, I, j = 0, \cdots, I - i\}$ we assume that the business has been observed during $I$ time periods. In [2], it is used, for practical situations, the rewritten version of the local linear estimator 3.1.1, using the regression formulation for aggregated data (in this thesis we consider that the data is aggregated in months). Let $z_{ij} = (x_i, x_j)$ be the coordinates of the data point $C_{i,j}$, $z_{ij}$ are points in the run-off grid and we assume that $C_{i,j}$ is the middle point of the corresponding area of a data cell in the triangle that contains the data. For density estimation in a discrete framework we have to extend the run-off grid to estimation grid with the points where the density will be estimated. The estimation grid $\nabla_{grid}$ is defined by its points as follows: the coordinates of density estimation points are denoted by $d_{kl} =$

$(d_k, d_l)$ such that

$$\nabla_{grid} = \{(d_k, d_l) \mid d_k = \Lambda(k - \frac{1}{2}), d_l = \Lambda(l - \frac{1}{2}), k = 1, \cdots, 2I, l = 1, \cdots, 2I\},$$

where $\Lambda$ is the bin length. The points in the estimation grid $d_{kl} \in \nabla_{grid}$ can be set as tightly as preferred but, as in [2], it will be considered that $\Lambda = \frac{1}{2}$ (see [2] for details and example with $I = 4$).

The local linear estimator for the density $f$ (with a support in a triangle) for any given point $d_{kl} = (x_k, y_l) \in \nabla_{grid}$ is given by solving the minimization problem:

$$\hat{\Theta} = \underset{\Theta}{\operatorname{argmin}} \sum_{(i,j)\in\nabla} \{C_{i,j} - \Theta_0 - \Theta_{1,1}(x_i - x_k) - \Theta_{1,2}(y_j - y_l)\}^2 K_{h_1}(x_i - x_k) K_{h_2}(y_j - y_l). \tag{3.1.2}$$

Here, the solution $\hat{\Theta}_0$ gives an estimator for $r(d_{kl})$ (the regression function). Then, the density $f(d_{kl})$ is estimated by

$$\hat{f}(d_{kl}) = \frac{\hat{r}(d_{kl})}{n\Lambda^2}, \qquad d_{kl} \in \nabla_{grid}.$$

Following the same ideas for giving the two-dimensional density in the complete square, we assume that $f(x_k, y_l) = f_1(x_k) f_2(y_l)$, applying the algorithm describes above changing the steps 2) and 3) by

2) Using $\hat{f}_1^{(0)}(x_k) = 1$ for every $x_k$,

$$\hat{f}_2^{(1)}(y_l) = \frac{\sum_{k=1}^{2I-l+1} \hat{f}(d_{kl})}{\sum_{k=1}^{2I-l+1} \hat{f}_1^{(0)}(x_k)}.$$

3) Using $\hat{f}_2^{(1)}(y_l)$, calculate the updated estimator for $f_1(x_k)$ as follows,

$$\hat{f}_1^{(1)}(x_k) = \frac{\sum_{l=1}^{2I-k+1} \hat{f}(d_{kl})}{\sum_{l=1}^{2I-k+1} \hat{f}_2^{(1)}(y_l)}.$$

and repeating the steps 2)-3) until the desired convergence criterion is achieved, we have the estimation for any point in the square in the case of aggregated data. In general, the initial value $\hat{f}_1^{(0)}$ can be taken as any constant function.

As we explained in the chapter 2, the choice of the kernel function is not as important as the choice of the value for the bandwidth. The choice of a kernel function does not usually affect the results considerably. The CCL model assumes a multiplicative structure, hence, multiplicative kernels are considered. As it was already said, we will consider the Epanechnikov kernel, which is often the default kernel function in most of the statistical software (see [2] to see popular kernels in multiplicative form and the efficiency of them relative to the Epanechnikov kernel). For the bandwidth selection

problem there exist several methods that try to solve it (see section 2.3.1 in Chapter 2). The main goal of this thesis is to analyze the effect of different bandwidths to the models and to find some rules of thumb about the choice of bandwidth under different assumptions about claim frequency, noise and other model parameters.

# Chapter 4

# Simulation study

We are carrying out a simulation study in this thesis, using the software R. Our interest is to model IBNR claims (monthly claims) and we consider in each simulation the parameters, claim frequency, average of a claim size and noise.

We simulated the complete rectangle of numbers of montly claims arrivals. In this simulation study we used Poisson distribution to generate the number of claims arrivals and the parameters $gencoef$ to determine the speed of claims development, where the value of parameter is between 0 and 1: high value indicates the claims to develop slow in the run-off triangle; low value of the given parameter indicates a fast development. We added noise to the generated claims using $sigma$ parameter (if $sigma$=0, no noise is added), $I$ is the number of periods (months) considered, $initClaimX$ denotes the initial value for the average number of claims per period (month). The number of claims are generated as

$$N_j + X_{sigma},$$

where $N_j \sim Po(initClaimX * (gencoef)^j)$ and $X_{sigma} \sim N(0, sigma)$. In the R program we consider the rounded values of $X_{sigma}$ to be sure that the number of claims generated are natural numbers.

Next step is to calculate the reserve estimate. First, we simulate the full distribution of IBNR claims claims over $I$ calendar periods. The advantage of using simulated data is that we can simulate data "to ultimate", and set aside the (otherwise unknown) losses at ultimate as a standard against which we can compare our model's predictions. Then, we use only the upper triangle to conduct the estimation with the CCL method. Lastly, when we have obtained the reserve estimates with the CCL method, we compare them with actual reserve estimates. The performance of the models is evaluated by comparing the reserve estimation errors using the mean absolute percentage error

$$MAPE(\hat{R}) = E\left(\frac{|R - \hat{R}|}{R}\right),$$

where $R$ denotes the outstanding liability estimated by a reserve estimate $\hat{R}$ at a given valuation date, this is

$$R = \sum_{i=2}^{I} \sum_{j=I-i+1}^{n-1} C_{i,j} \text{ and } \hat{R} = \sum_{i=2}^{I} \sum_{j=I-i+1}^{I-1} \hat{C}_{i,j},$$

where $\hat{C}_{i,j}$ denotes the actual values and $C_{i,j}$ the estimated values. The goal of this simulation study is to find out how the different simulation parameters along with the selected bandwidths affect the estimation precision. The parameters that are considered are the following: $m$ (the number of simulations considered), $I$ (number of periods (months)), $gencoef$ (determines the speed of claim's development), $sigma$ (to adjust the noise). In this study we consider $gencoef \in \{0.95, 0.9, 0.8\}$, $initClaimX \in \{50, 100, 400\}$ and $sigma \in \{0, 10, 100\}$. We consider only simulations with $I \in \{24, 48\}$ and $m = 100$ due to the computational reasons.

By previous experiences of other researchers using cross-validation we know that the optimal bandwidth $(h_1, h_2)$ in the CCL usually satisfied that $h_1 > h_2$. As cross-validation is time consuming, in this thesis we did not use cross-validation at all, we considered the simulations of all possible combinations of the parameters using some bandwidths (the choice of these bandwidth values was done randomly) $(h_1 = 10, h_2 = 5), (h_1 = 7, h_2 = 3.7)$, $(h_1 = 2, h_2 = 1)$ and $(h_1 = 1.2, h_2 = 0.7)$ and compared the results trying to find the effect of the parameters for the best bandwith among this specific choice of banwidths. From now on, when we use the expression "optimal bandwith" in this chapter, we are meaning the best bandwith among these specific banwidths.

The results obtained for $I = 24$ are given in Table 4.1 and for $I = 48$ in Table 4.2. The values in green are the smallest obtained MAPE values among the 4 sets of bandwidths. It indicates the best choice among the considered bandwidths in the simulation study. While referring to the parameters, we used the following notation:

$$gencoef; n; initClaimX; sigma.$$

For example, for $0.8; 24; 50; 0$, the optimal bandwidth value for the simulated trials is $(h_1 = 1.2, h_2 = 0.7)$. The value in red shows the highest MAPE value among the 4 fixed sets of bandwidths.

We saw that choosing big values for $I$ is time consuming, we could expect that if we spent 15 minutes for generating $100$ datasets with $I = 24$, then for simulating the same amount of data with the same parameters with $I = 48$ could be twice the time. But in reality it was not like this, it took for a computer sometimes between 5 and 8 hours with $I = 48$.

We observed that the majority of the optimal selections of the bandwidth for $I = 48$ are closer to the "reality" (as at the beginning we generated the complete rectangle we can consider that the lower triangle of this rectangle are future values that we already know) than for $I = 24$, it is the optimal bandwidth found with the other fixed parameters that has a MAPE smaller when $I = 48$, but the difference is not too big. An exception is found for the parameters $0.8; 100; 10$. Also, it is observed that for the parameter values $0.8; 50; 0$, $0.8; 100; 0$, $0.8; 400; 0$, $0.8; 400; 10$, $0.9; 50; 0$, $0.9; 50; 100$, $0.9; 400; 0$, $0.95; 100; 100$ the change of $I$ did not change the choice of the optimal bandwidth, and from this 8 parameter combinations 5 of them had as optimal bandwidth $(h_1 = 1.2, h_2 = 0.7)$.

The MAPE value of the optimal bandwidth with $sigma = 0$ is smaller than the corresponding for $sigma = 10$ and the last is smaller than the corresponding for $sigma = 100$. Also it is observed that for the parameters $gencoef = 0.8, sigma = 0$, we got that the $(h_1 = 1.2, h_2 = 0.7)$ values do not change for $I \in \{24, 48\}$ and $initClaimX \in \{50, 100, 400\}$. If we observed the data considering that the value of $initClaimX$ changes and the other parameters are fixed, we obtained that in all the simulations, MAPE value of the optimal bandwidth with $initClaimX = 50$ is bigger than the corresponding for $initClaimX = 100$ and the last is bigger than the corresponding for $initClaimX = 400$ (see Table 4.3 and Table 4.4).

Finally, we observed the data considering that the value of $gencoef$ is changing and the other parameters are fixed. We obtained for almost all the simulations that the MAPE value of the optimal bandwidth with $gencoef = 0.95$ is smaller than the corresponding value for $gencoef = 0.9$ and the value for value for $gencoef = 0.9$ is smaller than the corresponding for $gencoef = 0.8$ (see Table 4.5 and Table 4.6).

In general we observed that the optimal bandwidth values $(h_1 = 1.2, h_2 = 0.7)$ appeared more often in our simulations.

As the claims are simulated randomly, then the MAPE values change if we do twice the simulation with the same parameters, but it did not change the choice of the best bandwidth at least in those simulations that we had time to do twice.

TABLE 4.1: Results for $I = 24$

| | | | MAPE | | | |
|---|---|---|---|---|---|---|
| gencoef | initClaimX | sigma | h1=10, h2=5 | h1=7, h2=3.7 | h1=2, h2=1 | h1=1.2, h2=0.7 |
| 0.8 | 50 | 0 | 0.05239153 | 0.04950009 | 0.05031823 | 0.04283571 |
| 0.8 | 50 | 10 | 0.1130467 | 0.1205555 | 0.12076147 | 0.10934454 |
| 0.8 | 50 | 100 | 0.2110623 | 0.2515607 | 0.22917599 | 0.22330274 |
| 0.8 | 100 | 0 | 0.04896509 | 0.03481668 | 0.03591951 | 0.03268953 |
| 0.8 | 100 | 10 | 0.07504927 | 0.08340853 | 0.07783647 | 0.08365175 |
| 0.8 | 100 | 100 | 0.1860499 | 0.2292519 | 0.20086716 | 0.19275778 |
| 0.8 | 400 | 0 | 0.04966032 | 0.02362223 | 0.02098084 | 0.01932486 |
| 0.8 | 400 | 10 | 0.05300417 | 0.03417279 | 0.03528973 | 0.03585865 |
| 0.8 | 400 | 100 | 0.1429019 | 0.1439266 | 0.12918006 | 0.14100178 |
| 0.9 | 50 | 0 | 0.03898546 | 0.04087841 | 0.0370854 | 0.03580557 |
| 0.9 | 50 | 10 | 0.07094376 | 0.083007 | 0.09998923 | 0.09769359 |
| 0.9 | 50 | 100 | 0.2314684 | 0.2107483 | 0.20975367 | 0.18500695 |
| 0.9 | 100 | 0 | 0.02652894 | 0.02806231 | 0.02430436 | 0.02796716 |
| 0.9 | 100 | 10 | 0.06042328 | 0.06025865 | 0.0565313 | 0.048984 |
| 0.9 | 100 | 100 | 0.1727693 | 0.1764435 | 0.19795467 | 0.16746475 |
| 0.9 | 400 | 0 | 0.01305924 | 0.01289301 | 0.01463703 | 0.01301537 |
| 0.9 | 400 | 10 | 0.01785943 | 0.01726466 | 0.01865221 | 0.01649384 |
| 0.9 | 400 | 100 | 0.09514138 | 0.1000631 | 0.10537459 | 0.10242955 |
| 0.95 | 50 | 0 | 0.0306518 | 0.03391098 | 0.02720126 | 0.02902589 |
| 0.95 | 50 | 10 | 0.06921831 | 0.06098155 | 0.071658 | 0.06757005 |
| 0.95 | 50 | 100 | 0.2015863 | 0.2034348 | 0.21543789 | 0.24146918 |
| 0.95 | 100 | 0 | 0.01989925 | 0.02272246 | 0.02282828 | 0.02096513 |
| 0.95 | 100 | 10 | 0.03517107 | 0.03771949 | 0.03753382 | 0.03593285 |
| 0.95 | 100 | 100 | 0.1628117 | 0.1623241 | 0.19989581 | 0.17025282 |
| 0.95 | 400 | 0 | 0.01138654 | 0.01052849 | 0.06929823 | 0.01178066 |
| 0.95 | 400 | 10 | 0.01402963 | 0.01380302 | 0.01283079 | 0.01370195 |
| 0.95 | 400 | 100 | 0.06981378 | 0.07175351 | 0.07601472 | 0.07354817 |

TABLE 4.2: Results for $I = 48$

| | | | MAPE | | | |
|---|---|---|---|---|---|---|
| gencoef | initClaimX | sigma | h1=10, h2=5 | h1=7, h2=3.7 | h1=2, h2=1 | h1=1.2, h2=0.7 |
| 0.8 | 50 | 0 | 0.05123699 | 0.04785876 | 0.04523905 | 0.041593352 |
| 0.8 | 50 | 10 | 0.08878973 | 0.09721086 | 0.0869612 | 0.090049155 |
| 0.8 | 50 | 100 | 0.13114406 | 0.1336883 | 0.13318084 | 0.122783186 |
| 0.8 | 100 | 0 | 0.05548114 | 0.0348827 | 0.03631337 | 0.02819702 |
| 0.8 | 100 | 10 | 0.09647795 | 0.08025852 | 0.08926725 | 0.077744276 |
| 0.8 | 100 | 100 | 0.11402357 | 0.1312216 | 0.12677362 | 0.113933871 |
| 0.8 | 400 | 0 | 0.05023926 | 0.022444 | 0.01629616 | 0.013947765 |
| 0.8 | 400 | 10 | 0.04195101 | 0.03663222 | 0.03930255 | 0.039581354 |
| 0.8 | 400 | 100 | 0.10847604 | 0.1042241 | 0.10296888 | 0.089351645 |
| 0.9 | 50 | 0 | 0.02515505 | 0.02721037 | 0.02582932 | 0.026186506 |
| 0.9 | 50 | 10 | 0.06930435 | 0.0574893 | 0.07612974 | 0.066008021 |
| 0.9 | 50 | 100 | 0.1511512 | 0.1216148 | 0.13332033 | 0.116086557 |
| 0.9 | 100 | 0 | 0.02088546 | 0.01809355 | 0.02186476 | 0.016307195 |
| 0.9 | 100 | 10 | 0.05519259 | 0.04466511 | 0.05215932 | 0.04600764 |
| 0.9 | 100 | 100 | 0.10551851 | 0.1024782 | 0.1089914 | 0.112870781 |
| 0.9 | 400 | 0 | 0.01017455 | 0.0079162 | 0.0093448 | 0.008362374 |
| 0.9 | 400 | 10 | 0.01986731 | 0.01683672 | 0.0195674 | 0.019453192 |
| 0.9 | 400 | 100 | 0.07122501 | 0.07732212 | 0.06180346 | 0.07569379 |
| 0.95 | 50 | 0 | 0.0227879 | 0.02144606 | 0.01822586 | 0.016777859 |
| 0.95 | 50 | 10 | 0.05144733 | 0.05150686 | 0.05013246 | 0.045056431 |
| 0.95 | 50 | 100 | 0.12209427 | 0.1258446 | 0.12334552 | 0.114103484 |
| 0.95 | 100 | 0 | 0.01408179 | 0.01439067 | 0.01361614 | 0.013666387 |
| 0.95 | 100 | 10 | 0.02851292 | 0.02617789 | 0.03111062 | 0.029363331 |
| 0.95 | 100 | 100 | 0.10313513 | 0.08955379 | 0.114944 | 0.092549139 |
| 0.95 | 400 | 0 | 0.00714146 | 0.00646308 | 0.00748951 | 0.006144965 |
| 0.95 | 400 | 10 | 0.00945515 | 0.01034975 | 0.01064544 | 0.010492822 |
| 0.95 | 400 | 100 | 0.05351058 | 0.0479374 | 0.04971626 | 0.046846448 |

TABLE 4.3: Results for $I = 24$

| | | | MAPE | | | |
|---|---|---|---|---|---|---|
| gencoef | initClaimX | sigma | h1=10, h2=5 | h1=7, h2=3.7 | h1=2, h2=1 | h1=1.2, h2=0.7 |
| 0.8 | 50 | 0 | 0.05239153 | 0.04950009 | 0.05031823 | 0.04283571 |
| 0.8 | 100 | 0 | 0.04896509 | 0.03481668 | 0.03591951 | 0.03268953 |
| 0.8 | 400 | 0 | 0.04966032 | 0.02362223 | 0.02098084 | 0.01932486 |
| 0.8 | 50 | 10 | 0.1130467 | 0.1205555 | 0.12076147 | 0.10934454 |
| 0.8 | 100 | 10 | 0.07504927 | 0.08340853 | 0.07783647 | 0.08365175 |
| 0.8 | 400 | 10 | 0.05300417 | 0.03417279 | 0.03528973 | 0.03585865 |
| 0.8 | 50 | 100 | 0.2110623 | 0.2515607 | 0.22917599 | 0.22330274 |
| 0.8 | 100 | 100 | 0.1860499 | 0.2292519 | 0.20086716 | 0.19275778 |
| 0.8 | 400 | 100 | 0.1429019 | 0.1439266 | 0.12918006 | 0.14100178 |
| 0.9 | 50 | 0 | 0.03898546 | 0.04087841 | 0.0370854 | 0.03580557 |
| 0.9 | 100 | 0 | 0.02652894 | 0.02806231 | 0.02430436 | 0.02796716 |
| 0.9 | 400 | 0 | 0.01305924 | 0.01289301 | 0.01463703 | 0.01301537 |
| 0.9 | 50 | 10 | 0.07094376 | 0.083007 | 0.09998923 | 0.09769359 |
| 0.9 | 100 | 10 | 0.06042328 | 0.06025865 | 0.0565313 | 0.048984 |
| 0.9 | 400 | 10 | 0.01785943 | 0.01726466 | 0.01865221 | 0.01649384 |
| 0.9 | 50 | 100 | 0.2314684 | 0.2107483 | 0.20975367 | 0.18500695 |
| 0.9 | 100 | 100 | 0.1727693 | 0.1764435 | 0.19795467 | 0.16746475 |
| 0.9 | 400 | 100 | 0.09514138 | 0.1000631 | 0.10537459 | 0.10242955 |
| 0.95 | 50 | 0 | 0.0306518 | 0.03391098 | 0.02720126 | 0.02902589 |
| 0.95 | 100 | 0 | 0.01989925 | 0.02272246 | 0.02282828 | 0.02096513 |
| 0.95 | 400 | 0 | 0.01138654 | 0.01052849 | 0.06929823 | 0.01178066 |
| 0.95 | 50 | 10 | 0.06921831 | 0.06098155 | 0.071658 | 0.06757005 |
| 0.95 | 100 | 10 | 0.03517107 | 0.03771949 | 0.03753382 | 0.03593285 |
| 0.95 | 400 | 10 | 0.01402963 | 0.01380302 | 0.01283079 | 0.01370195 |
| 0.95 | 50 | 100 | 0.2015863 | 0.2034348 | 0.21543789 | 0.24146918 |
| 0.95 | 100 | 100 | 0.1628117 | 0.1623241 | 0.19989581 | 0.17025282 |
| 0.95 | 400 | 100 | 0.06981378 | 0.07175351 | 0.07601472 | 0.07354817 |

TABLE 4.4: Results for $I = 48$

| | | | MAPE | | | |
|---|---|---|---|---|---|---|
| gencoef | initClaimX | sigma | h1=10, h2=5 | h1=7, h2=3.7 | h1=2, h2=1 | h1=1.2, h2=0.7 |
| 0.8 | 50 | 0 | 0.05123699 | 0.04785876 | 0.04523905 | 0.041593352 |
| 0.8 | 100 | 0 | 0.05548114 | 0.0348827 | 0.03631337 | 0.02819702 |
| 0.8 | 400 | 0 | 0.05023926 | 0.022444 | 0.01629616 | 0.013947765 |
| 0.8 | 50 | 10 | 0.08878973 | 0.09721086 | 0.0869612 | 0.090049155 |
| 0.8 | 100 | 10 | 0.09647795 | 0.08025852 | 0.08926725 | 0.077744276 |
| 0.8 | 400 | 10 | 0.04195101 | 0.03663222 | 0.03930255 | 0.039581354 |
| 0.8 | 50 | 100 | 0.13114406 | 0.1336883 | 0.13318084 | 0.122783186 |
| 0.8 | 100 | 100 | 0.11402357 | 0.1312216 | 0.12677362 | 0.113933871 |
| 0.8 | 400 | 100 | 0.10847604 | 0.1042241 | 0.10296888 | 0.089351645 |
| 0.9 | 50 | 0 | 0.02515505 | 0.02721037 | 0.02582932 | 0.026186506 |
| 0.9 | 100 | 0 | 0.02088546 | 0.01809355 | 0.02186476 | 0.016307195 |
| 0.9 | 400 | 0 | 0.01017455 | 0.0079162 | 0.0093448 | 0.008362374 |
| 0.9 | 50 | 10 | 0.06930435 | 0.0574893 | 0.07612974 | 0.066008021 |
| 0.9 | 100 | 10 | 0.05519259 | 0.04466511 | 0.05215932 | 0.04600764 |
| 0.9 | 400 | 10 | 0.01986731 | 0.01683672 | 0.0195674 | 0.019453192 |
| 0.9 | 50 | 100 | 0.1511512 | 0.1216148 | 0.13332033 | 0.116086557 |
| 0.9 | 100 | 100 | 0.10551851 | 0.1024782 | 0.1089914 | 0.112870781 |
| 0.9 | 400 | 100 | 0.07122501 | 0.07732212 | 0.06180346 | 0.07569379 |
| 0.95 | 50 | 0 | 0.0227879 | 0.02144606 | 0.01822586 | 0.016777859 |
| 0.95 | 100 | 0 | 0.01408179 | 0.01439067 | 0.01361614 | 0.013666387 |
| 0.95 | 400 | 0 | 0.00714146 | 0.00646308 | 0.00748951 | 0.006144965 |
| 0.95 | 50 | 10 | 0.05144733 | 0.05150686 | 0.05013246 | 0.045056431 |
| 0.95 | 100 | 10 | 0.02851292 | 0.02617789 | 0.03111062 | 0.029363331 |
| 0.95 | 400 | 10 | 0.00945515 | 0.01034975 | 0.01064544 | 0.010492822 |
| 0.95 | 50 | 100 | 0.12209427 | 0.1258446 | 0.12334552 | 0.114103484 |
| 0.95 | 100 | 100 | 0.10313513 | 0.08955379 | 0.114944 | 0.092549139 |
| 0.95 | 400 | 100 | 0.05351058 | 0.0479374 | 0.04971626 | 0.046846448 |

TABLE 4.5: Results for $I = 24$

| | | | MAPE | | | |
|---|---|---|---|---|---|---|
| gencoef | initClaimX | sigma | h1=10, h2=5 | h1=7, h2=3.7 | h1=2, h2=1 | h1=1.2, h2=0.7 |
| 0.8 | 50 | 0 | 0.05239153 | 0.04950009 | 0.05031823 | 0.04283571 |
| 0.9 | 50 | 0 | 0.03898546 | 0.04087841 | 0.0370854 | 0.03580557 |
| 0.95 | 50 | 0 | 0.0306518 | 0.03391098 | 0.02720126 | 0.02902589 |
| 0.8 | 100 | 0 | 0.04896509 | 0.03481668 | 0.03591951 | 0.03268953 |
| 0.9 | 100 | 0 | 0.02652894 | 0.02806231 | 0.02430436 | 0.02796716 |
| 0.95 | 100 | 0 | 0.01989925 | 0.02272246 | 0.02282828 | 0.02096513 |
| 0.8 | 400 | 0 | 0.04966032 | 0.02362223 | 0.02098084 | 0.01932486 |
| 0.9 | 400 | 0 | 0.01305924 | 0.01289301 | 0.01463703 | 0.01301537 |
| 0.95 | 400 | 0 | 0.01138654 | 0.01052849 | 0.06929823 | 0.01178066 |
| 0.8 | 50 | 10 | 0.1130467 | 0.1205555 | 0.12076147 | 0.10934454 |
| 0.9 | 50 | 10 | 0.07094376 | 0.083007 | 0.09998923 | 0.09769359 |
| 0.95 | 50 | 10 | 0.06921831 | 0.06098155 | 0.071658 | 0.06757005 |
| 0.8 | 100 | 10 | 0.07504927 | 0.08340853 | 0.07783647 | 0.08365175 |
| 0.9 | 100 | 10 | 0.06042328 | 0.06025865 | 0.0565313 | 0.048984 |
| 0.95 | 100 | 10 | 0.03517107 | 0.03771949 | 0.03753382 | 0.03593285 |
| 0.8 | 400 | 10 | 0.05300417 | 0.03417279 | 0.03528973 | 0.03585865 |
| 0.9 | 400 | 10 | 0.01785943 | 0.01726466 | 0.01865221 | 0.01649384 |
| 0.95 | 400 | 10 | 0.01402963 | 0.01380302 | 0.01283079 | 0.01370195 |
| 0.8 | 50 | 100 | 0.2110623 | 0.2515607 | 0.22917599 | 0.22330274 |
| 0.9 | 50 | 100 | 0.2314684 | 0.2107483 | 0.20975367 | 0.18500695 |
| 0.95 | 50 | 100 | 0.2015863 | 0.2034348 | 0.21543789 | 0.24146918 |
| 0.8 | 100 | 100 | 0.1860499 | 0.2292519 | 0.20086716 | 0.19275778 |
| 0.9 | 100 | 100 | 0.1727693 | 0.1764435 | 0.19795467 | 0.16746475 |
| 0.95 | 100 | 100 | 0.1628117 | 0.1623241 | 0.19989581 | 0.17025282 |
| 0.8 | 400 | 100 | 0.1429019 | 0.1439266 | 0.12918006 | 0.14100178 |
| 0.9 | 400 | 100 | 0.09514138 | 0.1000631 | 0.10537459 | 0.10242955 |
| 0.95 | 400 | 100 | 0.06981378 | 0.07175351 | 0.07601472 | 0.07354817 |

TABLE 4.6: Results for $I = 48$

| | | | MAPE | | | |
|---|---|---|---|---|---|---|
| gencoef | initClaimX | sigma | h1=10, h2=5 | h1=7, h2=3.7 | h1=2, h2=1 | h1=1.2, h2=0.7 |
| 0.8 | 50 | 0 | 0.05123699 | 0.04785876 | 0.04523905 | 0.041593352 |
| 0.9 | 50 | 0 | 0.02515505 | 0.02721037 | 0.02582932 | 0.026186506 |
| 0.95 | 50 | 0 | 0.0227879 | 0.02144606 | 0.01822586 | 0.016777859 |
| 0.8 | 100 | 0 | 0.05548114 | 0.0348827 | 0.03631337 | 0.02819702 |
| 0.9 | 100 | 0 | 0.02088546 | 0.01809355 | 0.02186476 | 0.016307195 |
| 0.95 | 100 | 0 | 0.01408179 | 0.01439067 | 0.01361614 | 0.013666387 |
| 0.8 | 400 | 0 | 0.05023926 | 0.022444 | 0.01629616 | 0.013947765 |
| 0.9 | 400 | 0 | 0.01017455 | 0.0079162 | 0.0093448 | 0.008362374 |
| 0.95 | 400 | 0 | 0.00714146 | 0.00646308 | 0.00748951 | 0.006144965 |
| 0.8 | 50 | 10 | 0.08878973 | 0.09721086 | 0.0869612 | 0.090049155 |
| 0.9 | 50 | 10 | 0.06930435 | 0.0574893 | 0.07612974 | 0.066008021 |
| 0.95 | 50 | 10 | 0.05144733 | 0.05150686 | 0.05013246 | 0.045056431 |
| 0.8 | 100 | 10 | 0.09647795 | 0.08025852 | 0.08926725 | 0.077744276 |
| 0.9 | 100 | 10 | 0.05519259 | 0.04466511 | 0.05215932 | 0.04600764 |
| 0.95 | 100 | 10 | 0.02851292 | 0.02617789 | 0.03111062 | 0.029363331 |
| 0.8 | 400 | 10 | 0.04195101 | 0.03663222 | 0.03930255 | 0.039581354 |
| 0.9 | 400 | 10 | 0.01986731 | 0.01683672 | 0.0195674 | 0.019453192 |
| 0.95 | 400 | 10 | 0.00945515 | 0.01034975 | 0.01064544 | 0.010492822 |
| 0.8 | 50 | 100 | 0.13114406 | 0.1336883 | 0.13318084 | 0.122783186 |
| 0.9 | 50 | 100 | 0.1511512 | 0.1216148 | 0.13332033 | 0.116086557 |
| 0.95 | 50 | 100 | 0.12209427 | 0.1258446 | 0.12334552 | 0.114103484 |
| 0.8 | 100 | 100 | 0.11402357 | 0.1312216 | 0.12677362 | 0.113933871 |
| 0.9 | 100 | 100 | 0.10551851 | 0.1024782 | 0.1089914 | 0.112870781 |
| 0.95 | 100 | 100 | 0.10313513 | 0.08955379 | 0.114944 | 0.092549139 |
| 0.8 | 400 | 100 | 0.10847604 | 0.1042241 | 0.10296888 | 0.089351645 |
| 0.9 | 400 | 100 | 0.07122501 | 0.07732212 | 0.06180346 | 0.07569379 |
| 0.95 | 400 | 100 | 0.05351058 | 0.0479374 | 0.04971626 | 0.046846448 |

# Chapter 5

# Conclusion

The CLM can be regarded as a structured histogram type estimator on a triangle, so it can be improved by the kernel density estimator. Kernel density estimators are sensitive to the choice of bandwidth, but the choice of a kernel function does not usually affect the results considerably, for this reason the selection of the bandwidth of the kernel estimator is a subject of considerable research.

The CCL is a generalization of the CLM. In the CCL it is considered that the data is recorded in continuous time, the critical point of the CCL method is the choice of the bandwidth. There exist several methods for finding an optimal bandwidth, but they are time consuming. In this thesis we studied how different input scenarios affect the optimal bandwidths of the CCL model.

The simulation results suggest that $n, initClaimX, sigma, gencoef$ variables affect the process to find the best bandwidth (for our four fixed values of bandwidth), but in some cases, for some special parameters, there are no changes. The changes could be better seen if we would have more bandwidth values for comparing. Unfortunately we could not include more values in this study due to the time consuming procedures in R and the limited time. We could also observe that a small value of bandwidth tends to be in general more suitable for different sets of input parameters.

We think that this is a study that can be taken as a basis of further works about bandwidth selection in CCL, which is necessary to increase the predictive power of the CCL.

# Bibliography

[1] M. Käärik, *Non-life insurance mathematics (MTMS.02.053)*,University of Tartu, Estonia. (Lecture notes)

[2] M. Käärik, L. Tee, *Loss reserving on different levels of data aggregation: chain-ladder vs. its continuous extension.* (Submitted, 2017)

[3] M. D. Martínez Miranda, J. P. Nielsen, S. Sperlich and R. Verrall, *Continous Chain Ladder: Reformulating and generalizing a classical insurance problem. Expert Systems with Applications*, 40(14), 2013, 5588-5603.

[4] J.P. Nielsen, *Multivariate bounded kernels from local linear estimation*, Scandinavian Actuarial Journal, 1, 1999, 93-95.

[5] M. P. Wand and M. C. Jones, *Kernel smoothing*. Crc. Press, 1994.

[6] M. V. Wüthrich, *New developments in claims reserving, Lecture 1: Mack's Chain-Ladder Model for Claims Reserving*, RiskLab. ETH Zurich, 6th. St. Petersburg spring school, March 23-25, 2017. (Lecture slides)

[7] W. Zucchini, Applied smoothing techniques, Part 1: Kernel density estimation, 2003, 2-19. Seen at http://staff.ustc.edu.cn/ zwp/teach/Math-Stat/kernel.pdf, january 25th 2017.