Open Source Licenses and Project Growth

Diplomarbeit im Fach Informatik

vorgelegt von

Gottfried Hofmann

geb. 16.09.1983 in Münchberg

angefertigt am

Department für Informatik
Professur für Open Source Software
Friedrich-Alexander-Universität Erlangen–Nürnberg

Betreuer: Prof. Dr. Dirk Riehle

Beginn der Arbeit: 21.09.2011
Abgabe der Arbeit: 21.03.2012

Ich versichere, dass ich die Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Der Universität Erlangen-Nürnberg, vertreten durch die Professur für Open Source Software, wird für Zwecke der Forschung und Lehre ein einfaches, kostenloses, zeitlich und örtlich unbeschränktes Nutzungsrecht an den Arbeitsergebnissen der Diplomarbeit einschließlich etwaiger Schutzrechte und Urheberrechte eingeräumt.

Erlangen, den

Gottfried Hofmann

Diplomarbeit

Thema: Open Source Licenses and Project Growth.

Hintergrund: It has long been hypothesized that the choice of an open source license impacts the growth of open source projects. This thesis analyses the Ohloh data set, a large database of active well-working open source projects, for the relationship between choice of license and project growth. It provides models for licenses and project growth and correlates them.

Aufgabenstellung:

- Prior work, review of literature
  Review of models of open source licenses
  Review of open source project growth
  Review of work correlating project growth with licenses
- Model of open source licenses
  Provision of a small model of licenses, e.g. permissive vs reciprocal
  Classification of existing licenses according to model
- Model of open source project growth
  Provision of several observable variables indicative of project growth
  Minimally, growth in lines of code, commit frequency, number of committers
  Provision of analytically closed models for these variables, binned by licenses
- Comparison of by-license community growth models and evaluation thereof
  Provision of confidence measure as to statistical significance of differences
- Discussion of causation vs. correlation in this context
  Discussion of other impacting factors, e.g. people behind license choice
  Discussion of un/desired effects of license on development and use situation

Betreuung: Prof. Dr. Dirk Riehle

Bearbeiter: Gottfried Hofmann

Table of Contents

1 Abstract 1

2 Introduction 2

2.1 Background 2
2.2 A short history of open source licensing 2
2.3 A model of permissive and restrictive open source licenses 4
2.4 Reasons to license permissive or restrictive and general trends 4

3 Open Source Project Growth 6

3.1 Measurements of project growth used in this thesis 6
3.2 Literature dealing with overall open source project growth 6

4 The Ohloh Database 7

4.1 Collection method and sample size 7
4.2 Cleanup process of the Ohloh data 7

5 A model for the total growth binned by licenses 13

5.1 A first glimpse at the cleaned data using LOESS for smoothing 13
5.2 Using self-starter function models in R to fit nonlinear models 13
5.3 A closer look at the exponential model 16
5.4 Transforming the response 18
5.4.1 Linear Regression on the log-transformed response 19
5.4.2 Segmenting the linear model 21
5.4.2.1 Ordinary Least-Squares Approach 21
5.4.2.2 Generalized Least-Squares approach 25
5.5 Discussion of the models 27
5.5.1 Non-segmented linear approach 27
5.5.2 Segmented linear approach 28
5.5.3 Segmented Linear Approach with Generalized Least-Squares 31
5.5.4 Discussion of confidence 32
5.6 The models transformed to normal scale 32

6 A model for the growth of active projects per month binned by licenses 36

6.1 A first glimpse at the number of projects using LOESS for smoothing 36
6.2 Using self-starter functions models in R to fit nonlinear models 36

6.3 A closer look at the exponential model 39
6.4 Log-transformation of the response 41
6.4.1 Linear regression on the log-transformed response 41
6.4.2 Segmenting the linear model 43
6.4.3 Linear regression of quadratic model on the log-transformed response 45
6.4.4 Segmentation of the quadratic model 47
7 Normalization of the added SLoC per month by number of active projects 50
7.1 A first glimpse at the data using Loess for smoothing 50
7.2 Using Self-starter functions in R to fit non-linear models 50
8 Discussion of results and impacting factors 52
9 Limitations of analysis 54
10 Conclusion and further research 55
10.1 Conclusion 55
10.2 Further Research 55
10.3 Acknowledgements 55
Abbreviations 56
Illustration Index 57
Index of Tables 62
Index of Literature 63
License 66

1 Abstract

What license to choose or change? That's a question many open source projects will face at least once. Besides philosophical reasons to favor one type of license over another there is the concern wether the chosen license has an impact on the projects success. But does the license really matter that much regarding the latter?
This thesis provides a two-bin model of open source licenses (permissive vs. restrictive) and analyses wether there is any impact on the growth of open source projects. The analysis is based on a sample of roughly 30% of all open source projects from the time period 1995-2007. Growth is determined by absolute growth of all projects, growth in number of active projects and average growth per project. Correlation is done by license type.
It can be shown that for a period from 1995 to roughly 2000/2002 there is a significant difference in the total growth in SLoC with the restrictive set growing faster. This changes for the time period from roughly 2000/2002 to 2007.

2 Introduction

The following chapter provides an introduction to the field of research of this thesis. Chapter 2.1 gives an introduction to the reasoning behind license choices. Chapter 2.2 provides a short history of open source licensing focusing on the two licensing paradigms 'permissive' and 'restrictive'. Chapter 2.3 provides a model of restrictive and permissive licenses based on the model proposed in earlier research. Chapter 2.4 gives and overview of literature that deals with reasons for choosing a certain type of license and recent trends in open source licensing.

2.1 Background

Research on open source software (OSS) and development processes has gained significant momentum over the last decade. Landmark work was published by Lerner and Tirole in 2003 [1]. A meta-study was conducted by Aksulu and Wade in 2010 [2] to give an overview of the state of the research in the field. Yet many basic questions remain to be answered. One of them is the question of licensing.
When a project has the ability to chose the license freely, such a choice can be controversial among developers. Same applies to the situation where a project decides to switch from one license to another.
The pros and cons for certain licenses can come from philosophical, ethical, pragmatic and many more standpoints. One factor is whether licensing has an impact on the possible growth of a project and if yes, which license type will allow for faster growth or a higher chance of survival in the OSS ecosystem.
This thesis analyses whether a correlation between the chosen license type and both the overall growth of OSS projects, the number of active projects and the average growth per project can be determined for the time period of 1995 to 2007.

2.2 A short history of open source licensing

In the early days of the computer era (roughly from the early 1960 to the early 1980s), sharing of source code for computer programs was commonplace and conducted in an informal manner. This kind of collaboration happened in an academic setting. When commercial companies started to enforce intellectual property rights, the first open source licenses emerged as an effort to retain the collaborative environment by providing philosophical and ethical grounds and a legal framework.
The first of these initiatives where the GNU project [3], which launched in 1983, and the Free Software Foundation (FSF) [4] in 1985 (both founded by Richard Stallman). The GNU project set the ethical and philosophical grounds, which where scripted in the GNU manifesto in 1985 [5].
The FSF started as an entity to support the development of Free Software by hiring software developers for the GNU project and as a copyright holder for the source code of said project. In 1986, a first version [6] of the Free Software Definition was published. The current version [7] lists the following four basic freedoms (counting from 0 to 3) for the licensee (bold formatting added for emphasis):

- The freedom to run the program, for any purpose (freedom 0).
- The freedom to study how the program works, and change it so it does your computing as you wish (freedom 1). Access to the source code is a precondition for this.
- The freedom to redistribute copies so you can help your neighbor (freedom 2).

- The freedom to distribute copies of your modified versions to others (freedom 3). By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.

The focus shifted towards providing a legal framework for developers who wanted to release programs under terms complying with the Free Software Definition with the release of the GNU General Public License (GPL) version 1 in 1989 [8][1]. The license soon was adopted by a variety of projects not related to GNU.
To assure freedom 1 and 3 for the licensee, the GPL includes a clause that forces developers who make changes to the code to release their changes under the same conditions as the GPL. This property of the GPL led to the attribution of the GPL as a 'viral' [9] or 'reciprocal' license. Another term for this kind of licensing is 'Copyleft' while software released under such terms is called 'Free Software'. For the remainder of this thesis, licenses of this kind will be called 'restrictive'.

In 1988, two licenses were first published who's conditions were later coined 'Copyfree' or 'permissive', namely the MIT license [10] and BSD license [11][2]. Both do not require derived work to be licensed under the same terms[3], thus using code for proprietary products is possible.

Later licenses were created like the GNU Lesser General Public License (LGPL) that are less restrictive than the GPL-like licenses yet still not completely permissive. Projects that use those licenses are not subject of this analysis for the sake of simplicity.

Note that both license types emerged roughly at the same time, so none of the two types used for the analysis here had a "head-start" over the others. Yet, restrictive licenses like the GPL happen to be more widely used up until today.

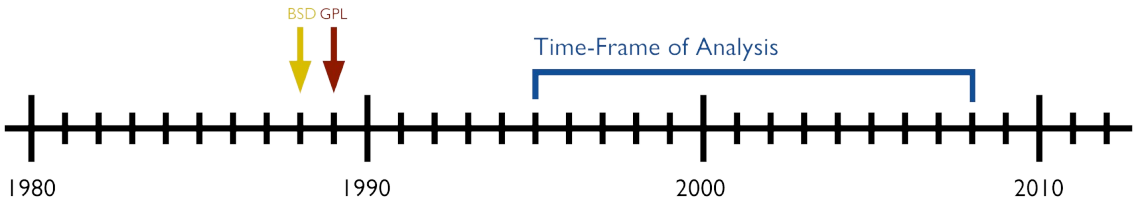

Illustration 1: Time bar of the analyzed time period and the introduction of the prototypes of the two license-types used for binning.

The term 'open source', which is used for both Free Software and permissive-licensed programs in this thesis, was coined in 1997 in an endeavor to formulate criteria for licensing of software that are less strict and ethics-oriented than the Free Software Definition and suitable for business use [12].
In 1998 the Open Source Initiative (OSI) was formed [13]. The OSI keeps a list of licenses that have been run through an approval process [14] that they are compatible with the open source definition.

1 Before the release of the GPL, various GNU projects used different licenses each tailored to the project but incompatible to each other in some cases. The GPL not only provided a unification of the licensing of the GNU project but also enabled developers not associated with the FSF to easily license their software by using the publicly available GPL template.
2 Both licenses are available in multiple versions now, like the 2-clause, 3-clause and 4-clause BSD license or the X11 license.
3 Yet there may still be restrictions like in the 'New BSD License' which does not permit advertising of derived products with the name of the licensor.

Both the OSI and FSF are advocates of open source software. While the FSF focuses on ethical issues with software licensing and describes itself as a social movement[4], the OSI is business-oriented, promoting open source as a method of cost-reduction, transparency and as a development-model[5]. While the FSF (with some exceptions) recommends the use of restrictive licenses[6], the OSI does not endorse a special type of license.

2.3 A model of permissive and restrictive open source licenses

The model for permissive and restrictive licenses in this thesis is based on the model proposed by Lerner and Tirole [17]. It was expanded by additional licenses that occur in the data set. The additional licenses are required to be approved by the OSI.

Permissive		Restrictive	
License Name	Observations in Sample	License Name	Observations in Sample
BSD	730	GPL	3248
MIT	378	CC-BY-SA	24
Apache	479		
zlib/libpng	26		
Public Domain	34		
Artistic License	210		
Python License	17		
Sun Industry Standards	-		
Zope	8		
Vovida	1		

Table 1: Licenses by Type. Multiple versions of a license are counted as one. For example GPL v1, v2 and v3 are listed as GPL only[7].

2.4 Reasons to license permissive or restrictive and general trends

In many situations, the choice of license of a project is predetermined. For example if the project is a fork of a project with restrictive license or if the project wants to built upon software components that are licensed restrictive. For the case where a projects is free to chose a suitable license, various studies have been conducted in the past to find out about the rational behind such a choice. Sen, Subramian and Nelson suggest that “OSS managers who want to attract a limited number of highly skilled programmers to their open source project should choose a restrictive OSS license. Similarly, managers of software projects for social programs could attract more developers by choosing a

4 “Open source is a development methodology; free software is a social movement.” [15]

5 “OSI Board members frequently travel the world to attend Open Source conferences and events, meet with open source developers and users, and to discuss with executives from the public and private sectors about how Open Source technologies, licenses, and models of development can **provide economic and strategic advantages**.” [13] (emphasis added).

6 “In the GNU Project we usually recommend people use copyleft licenses like GNU GPL, rather than permissive non-copyleft free software licenses. We don't argue harshly against the non-copyleft licenses—in fact, we occasionally recommend them in special circumstances [...]” [16]

7 The Observations in sample where counted **after** the cleanup process described in chapter 4.2

restrictive OSS license." [18]. Lerner and Tirole argue that "Projects with unrestricted licenses attract more contributors." [17]. In contrast, Colazo and Fang (2009) [19] analyzed 44 restrictive- and 18 permissive-licensed projects from the SourceForge database. The restrictive-licensed projects had a significantly higher developer membership and coding activity.
In a series of articles [20] [21] [22], Aslett (2011) describes a recent trend in open source licensing that the ration of permissive- vs. restrictive-licensed projects is slowly shifting in favor of permissive licensing. Source for the data is both the Ohloh database and FLOSSmole. Note that these findings can not be verified in this thesis as he looks at very recent trends from 2008 onwards while the snapshot of the Ohloh database used in this thesis proved to be reliable only up until the middle of 2007.

3 Open Source Project Growth

The following chapter deals with project growth in open source research. Chapter 3.1 describes the measurements of growth used in this thesis. Chapter 3.2 gives an overview of literature dealing with project growth of open source software in general (not binned by licenses).

3.1 Measurements of project growth used in this thesis

As a measurement of project growth, this thesis uses the metric Source Lines of Code (SLoC) added per month. A SLoC is a line in a commit that is neither empty nor a comment. Herraiz, Gonzalez-Barahona and Robles [23] have compared SLoC to various other common metrics of size and complexity of software projects and found a high correlation between all chosen metrics. This and the overall usage of SLoC in software engineering literature make it good candidate as a metric for project growth.
The lines for SLoC-calculation are calculated using the Unix Diff-Command between two consecutive versions. Then the SLoC are are binned for each project in a month-window.
As a second metric, SLoC per month of all projects are normalized by the number of active projects in the sample, resulting in the average growth-per-project. Whether a project is considered active in a given month is calculated using the following metric:

A project is considered active if the number of commits during the last 12 month amounted for at least 60% of the commits of the prior 12 months. A project not active at a given time is considered dead.

3.2 Literature dealing with overall open source project growth

Among the literature dealing with open source project growth, different metrics and sample sizes are employed. Deshpande and Riehle (2008) [24] use 5122 active and popular open source projects from the Ohloh database as a sample[8] and find that open source in both added SLoC per month and new projects per month shows in total exponential growth. Several studies have analyzed individual projects and small samples and discovered sub-linear[28], linear [25][29][30] and super-linear [25] [26][27] growth-per-project. Vasa (2010) [31] argues that the growth of an open source project can be super-linear over its entire lifespan but still show periods of linear or sub-linear growth, suggesting a segmented growth-pattern when analyzing single projects. A large-scale study by Koch [32][33] on projects from the Sourceforge database suggests that small projects in general show linear growth and that the chances for super-linear growth are positively related with the size of a project.

8 An earlier snapshot of the same database used in this thesis.

4 The Ohloh Database

The following chapter describes the sample source used for the analysis – the Ohloh database. Chapter 4.1 describes how the data was collected, how detailed it is and for what percentage of the overall open source projects data is available. Chapter 4.2 describes what data was used for the analysis and how it was cleaned from adulerant factors like projects switching repositories, projects switching from closed-source to an open source development model and outliers from projects importing large chunks of code from other projects.

4.1 Collection method and sample size

The Ohloh database has been collecting data of open source projects since 2005. It currently holds statistics about an estimate of 30% of all open source projects. A project is added to the database either if it's suggested by an individual in a Wiki-like process or if Ohloh considers the project popular. Popularity is measured by the number of in-links from the Yahoo! search engine. The provided data is collected from publicly visible revision control repositories (SVN, CVS, Git etc.) on a weekly basis and includes low-level data like individual developer's actions. According to Koch (2005), revision control systems are a very good source to study open source projects:

> „In open source software development projects, repositories in several forms are also in use, in fact form the most important communication and coordination channels, as the participants in any project are not collocated. Therefore only a small amount of information can not be captured by repository analyses because it is transmitted inter-personally. As a side effect, the repositories in use must be available openly and publicly, in order to enable as many persons as possible to access them and to participate in the project. Therefore open source software development repositories form an optimal data source for studying the associated type of software development.“ [34].

Since most revision control systems provide a history, the Ohloh database includes data from as early as 1983 [35]. For this thesis, data from before 1995 was omitted because it proved to be too sparse and unreliable to be of any use.

4.2 Cleanup process of the Ohloh data

The Ohloh database snapshot used in this thesis includes a table called 'activity facts' which should provide the required data[9]. Nevertheless it proved to be very unreliable when testing it against real data of the Linux kernel. Thus a method was developed to create an own version of the activity from the raw Ohloh data which proved to be way more realistic after a few adjustments had been made. Illustration 2 and illustration 3 show the raw added SLoC for the permissive and restrictive data sets without any cleanup applied[10].

9 An up-to-date version is publicly available through the Ohloh API [36], which is limited to 1.000 requests per day.

10 To make spotting noise and peaks easier the total added lines of code are formatted to a total of 6 Million in the restrictive sample and 3 Million in the permissive one.

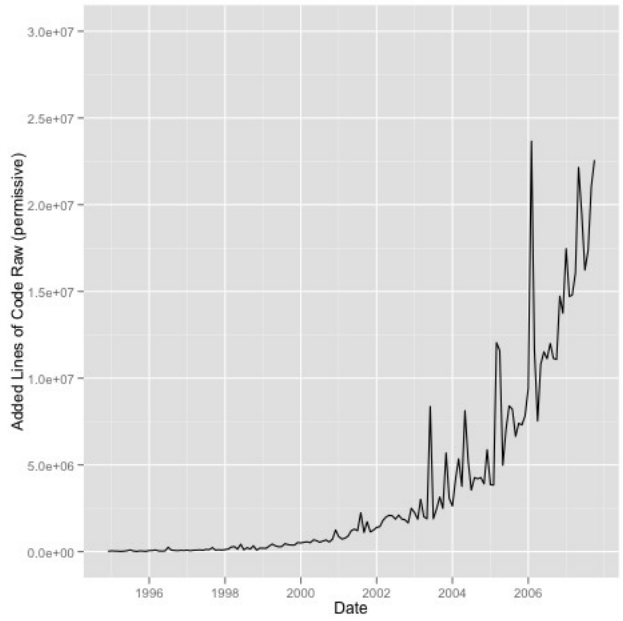

Illustration 2: Raw SLoC added permissive

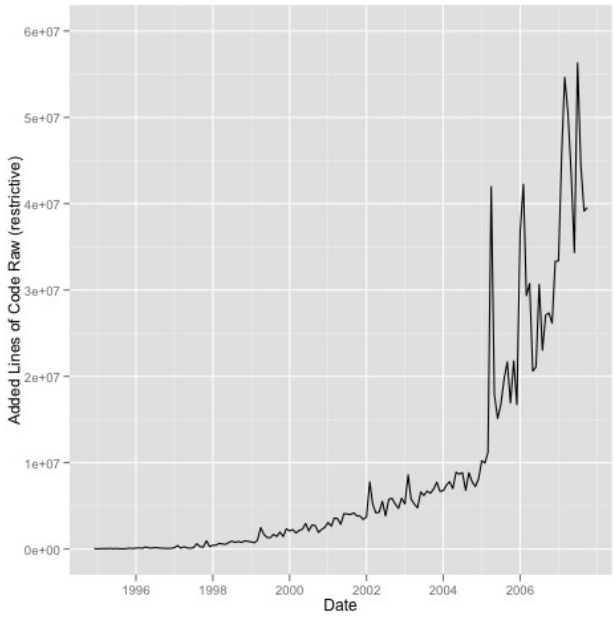

Illustration 3: Raw SLoC added restrictive

The Oholo database does store low-level information about each project, including the repository used. In the case of a project switching from one repository to another, both repositories are commonly used for a certain time frame. During that time frame, the data gathered by Ohloh accounts for both repositories, which means there is a high chance of duplicates.
To tackle the problem of projects running multiple repositories in parallel, a cleaning algorithm was employed. If a projects uses two repositories with the same content, the duplicates where discarded[11]. If a project runs multiple different repositories over the entire lifespan, the contents where combined into one data set[12]. For the rest of the projects which list multiple repositories, a heuristic was employed.
For the months with multiple repositories, the sum, mean, maximum and minimum of all SLoC added in that month where computed. Those where compared against the last month with only one listed repository before the period of multiple repositories[13]. The data of that 'previous month' was compared against the aggregates in the following way: If the sum of the aggregated data was up to twice as high as the added lines of code of the 'previous month' the sum was used, otherwise if the mean was up to twice as high, the mean was returned, otherwise the same procedure was conducted with the maximum and the minimum. If all those metrics were more than twice as high as the added lines of code of the previous month, the mean of added lines of code of all projects was used.

Illustration 4 and 5 show the added SLoC per month-window after the issue of projects using multiple repositories was accounted for.

11 280 project-month-tuples where discarded this way from the 'permissive' dataset and 642 from the 'restrictive' dataset
12 This was the case with 31 projects in the 'permissive' dataset and 39 from the 'restrictive' dataset
13 If there is no such date, the first month after the period of multiple repositories was used. Since projects which list multiple repositories for their entire timespan had already been taken care of, these two should be the only possibilities left.

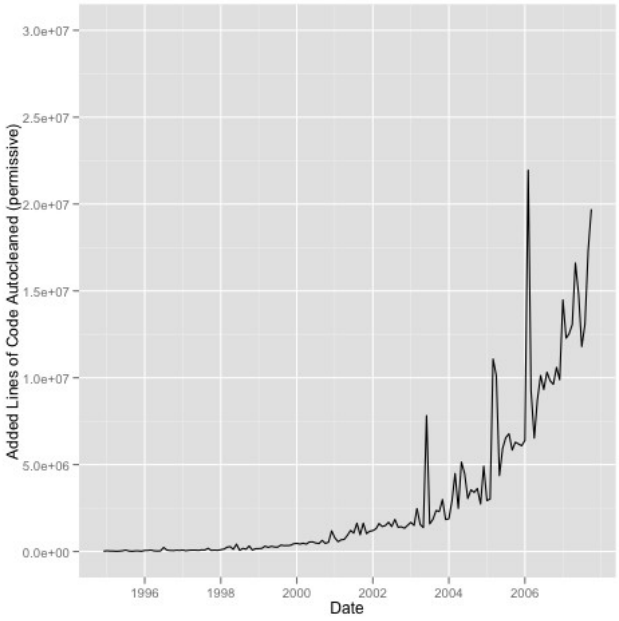

Illustration 4: Added SLoC without multiple repositories permissive.

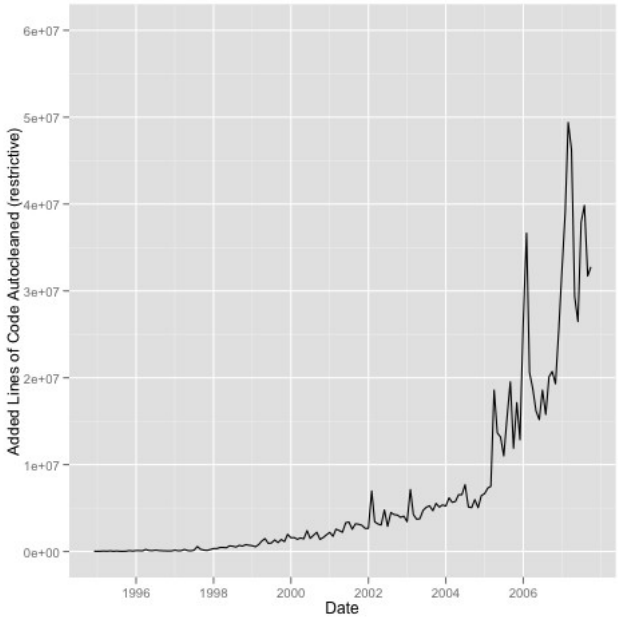

Illustration 5: Added SLoC without multiple repositories restrictive.

Open source projects don't always start open source. Sometimes a project starts proprietary and is released as open source at a later point or a team starts working on a project in a private fashion and opens the development process when it has reached a mature state. In case of a fork, the first commit is mostly a full copy of an existing project[14].
Further, to measure growth, the size at 'birth' of a project is not of interest and thus the initial commit[15] of each project was removed[16]. The resulting datasets are plotted in Illustration 6 and 7.

14 Note that this does not account for the case when a project becomes Open Source but the history of the revision control system is preserved or when a fork imports the history, too.
15 Actually the first month of commits.
16 This also eliminates projects that ceased development after the initial commit.

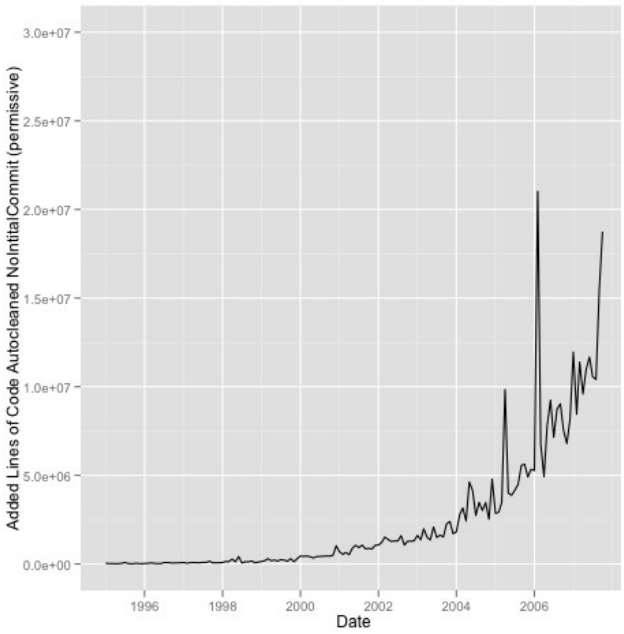

Illustration 6: Added SLoC without multiple repositories and initial commit (permissive).

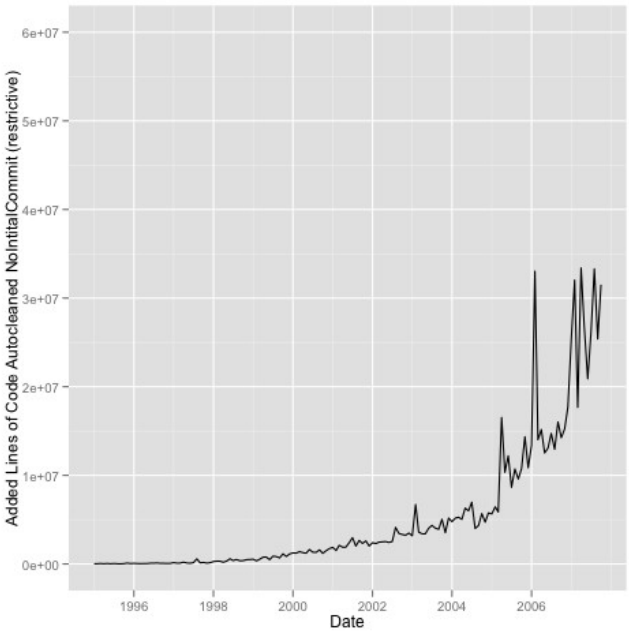

Illustration 7: Added SLoC without multiple repositories and initial commit (restrictive).

The resulting data still showed various outliers. A manual inspection revealed that most of those outliers came from projects importing code from other projects like from the Linux Kernel. Thus a manual analyzation of the most visible outliers was conducted. Since outliers where identified visually, a normalization by the number of active projects[17] was conducted beforehand. Otherwise outliers in the earlier years might have stayed unnoticed due to the heteroscedasticity. Illustration 8 and 9 show the normalized data used for outlier identification.

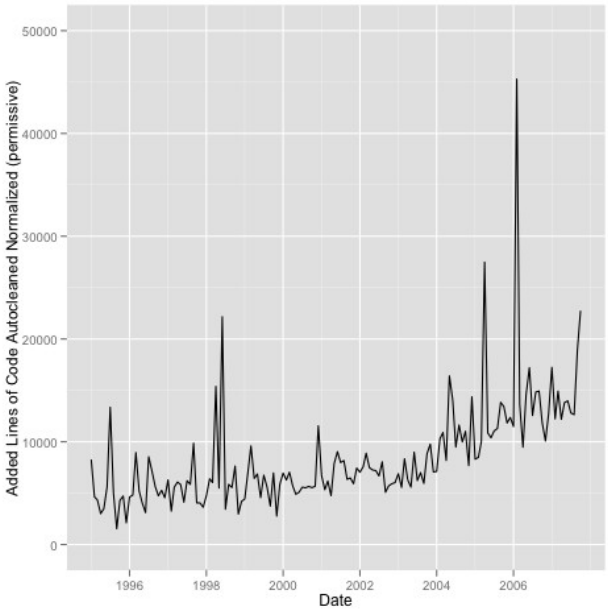

Illustration 8: Normalized data for outlier detection (permissive)

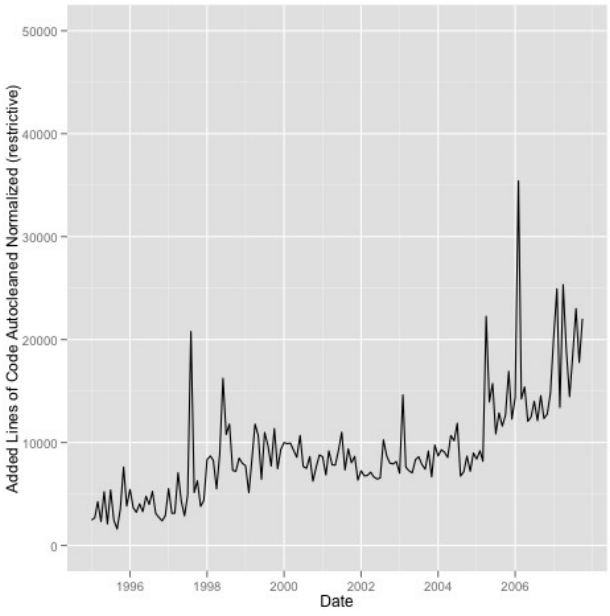

Illustration 9: Normalized data for outlier detection (restrictive)

17 For the definition of the activity state of a project in this thesis see 3.1.

Projects where removed from the set for various reasons. Two projects (Android and BRL-CAD) had data in the set spanning years before they where released as open source. A proprietary development-history is not of interest when studying the growth of open source software. Two projects where compilations of tools that had added the entire Linux-kernel to their repository. One project was a 1:1 fork of another project without own additions for the span of several months. For one project (GCC) a month was discovered in which several sub-projects got merged into one. One project (ReactOS) had a month of extensive code-review in which the entire codebase got re-commited. One project (Tcl/Tk) had an unusual high amount of code commit in one month but no reason was found for that[18].

Illustration 10 and 11 show the data after the cleanup process was finished:

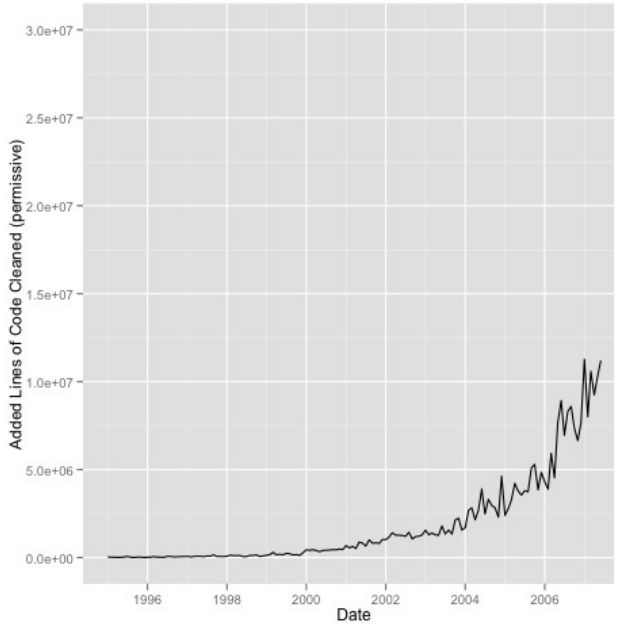

Illustration 10: Cleaned data (permissive)

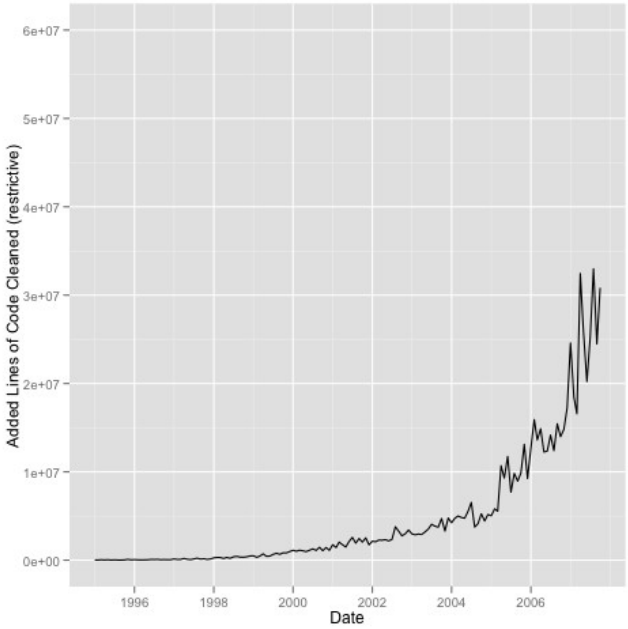

Illustration 11: Cleaned data (restrictive)

Illutration 12 and 13 show the data normalized by number of active projects:

18 For further details on the projects and months that got removed consult the R-script 'CleanDataManual.R', which can be found in the source-repository for this thesis.

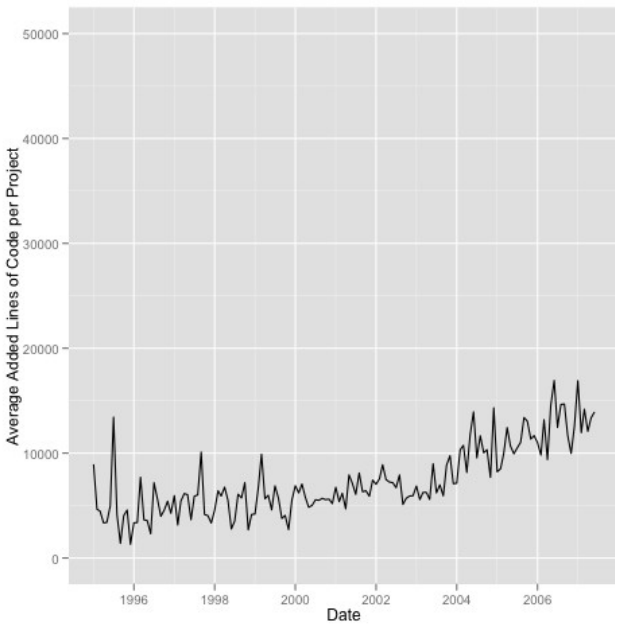

Illustration 12: Average added lines of code per project (permissive)

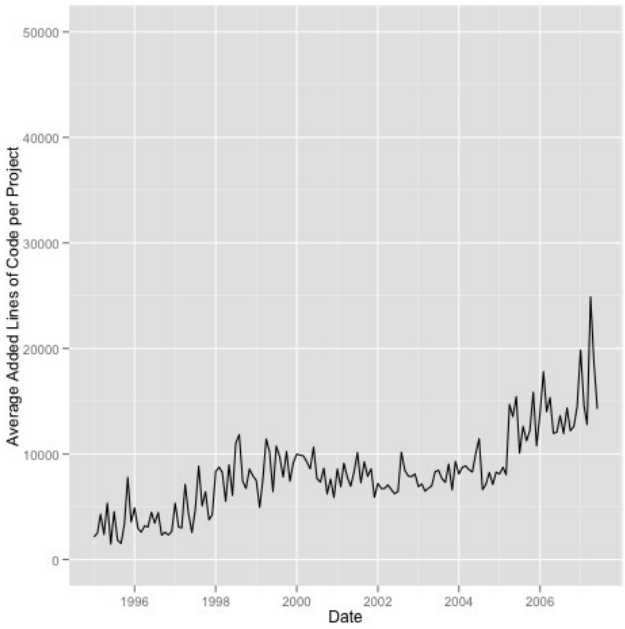

Illustration 13: Average added lines of code per project (restrictive)

After the cleanup process, the sample constituted of 1861 projects in the category 'permissive' and 3257 projects in the category 'restrictive'.

5 A model for the total growth binned by licenses

The following chapter describes the various steps taken towards an analytically closed model for the total growth binned by licenses. Chapter 5.1 describes how a Loess-curve was used to get a first idea of the underlying trend. Chapter 5.2 describes how self-starter functions for non-linear models in R where used to find a model with a good fit of the the data. Chapter 5.3 describes how the model with the best Goodness-of-Fit (GoF) was analyzed for model violations. Chapter 5.4 describes a series of remedies for the discovered model violations by applying a log-tranformation to the response and using various linear approaches to fit a model to the data. Chapter 5.5 discusses the approaches and their results regarding the research question. Chapter 5.6 shows the discovered models re-transformed to normal scale.

5.1 A first glimpse at the cleaned data using LOESS for smoothing

The cleaned data was plotted against a Loess curve to gain a first insight of the underlying trend (Illustration 14 and 15). As opposed to the last chapter the scale of the y-axis of both plots is the same.

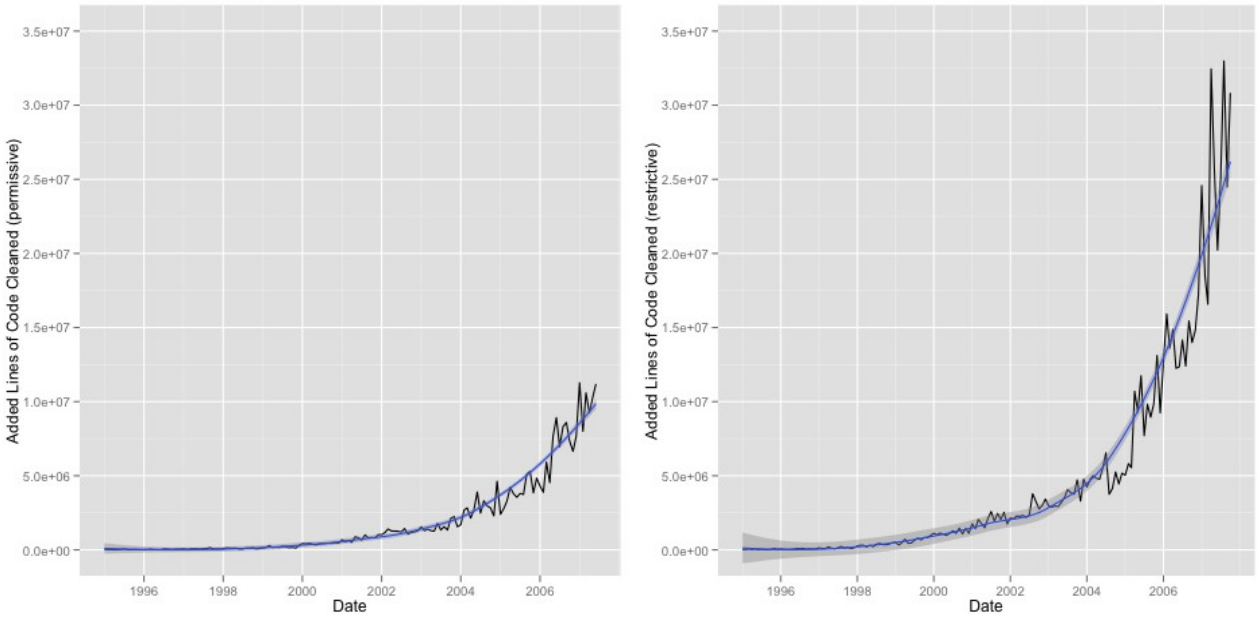

Illustration 14: Cleaned data with Loess curve in blue (permissive)

Illustration 15: Cleaned data with Loess curve in blue (restrictive)

The plot suggests a non-linear model for the growth, possibly logistic or exponential[19].

5.2 Using self-starter function models in R to fit nonlinear models

To find a fitting regression model, the R-function 'nls' was used in conjunction with self-starter functions for various models for non-linear growth. In addition, quadratic and cubic functions where tested. Table 2 lists the tested models. When a regression was successful[20], the Goodness-of-Fit using Pearson's r^2 for each model is also listed.

19 The latter as suggested by Deshpande and Riehle (2008) [24]

20 Some of the models don't make sense for this case but later on for other regressions. Since the process of fitting self-starter-models was semi-automated for this thesis, the list of models was re-used.

Model (name of self-starter-function in R)	Goodness-of-Fit (Pearson's r^2)	
	Permissive	Restrictive
SSmicmen	-	-
SSbiexp	-	-
SSasymp	-	-
SSasympOff	-	-
SSasympOrig	-	-
SSgompertz	-	-
SSflp	-	-
SSlogis	-	-
SSweibull	-	-
Quadratic	0.8962546	0.8604079
Qubic	0.9483102	0.9210685
SSexp[21]	0.9602230	0.9366640

Table 2: List of models tried for total growth with Goodness-of-Fit binned by license.

The quadratic, cubic and exponential models fit with a GoF > 0.9, which can be considered 'good'. To further investigate whether the models suitably represent the growth, each model was plotted against both the data and Loess curve. Illustration 16 and 17 show the quadratic model:

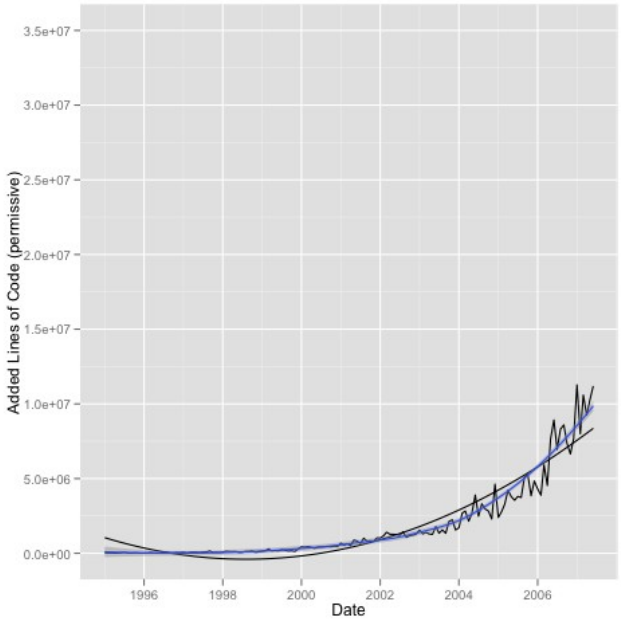

Illustration 16: Quadratic model against overall added SLoC (permissive)

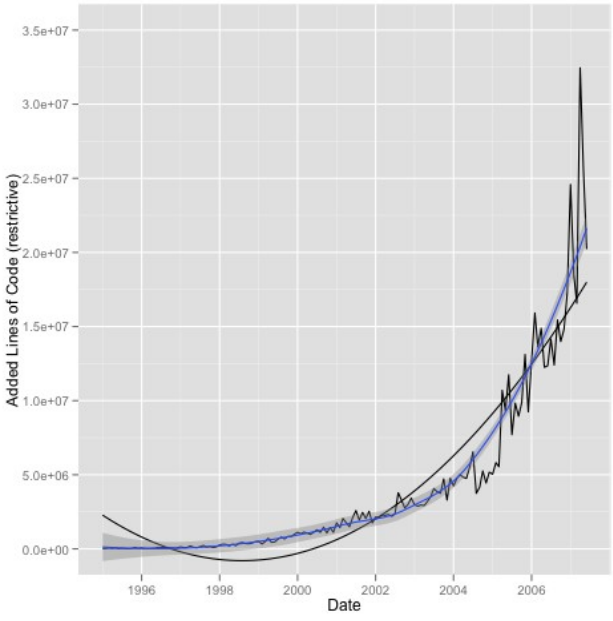

Illustration 17: Quadratic model against overall added SLoC (restrictive)

Visually, the model fails to explain the trend at all dates.

21 Not part of the default R installation but from the package 'nlrwr'

Illustration 18 and 19 show the cubic model:

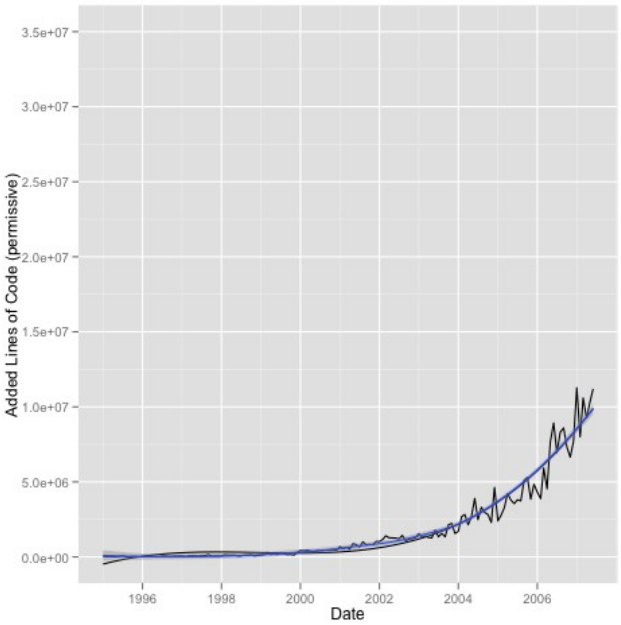

Illustration 18: Cubic model against overall added SLoC (permissive)

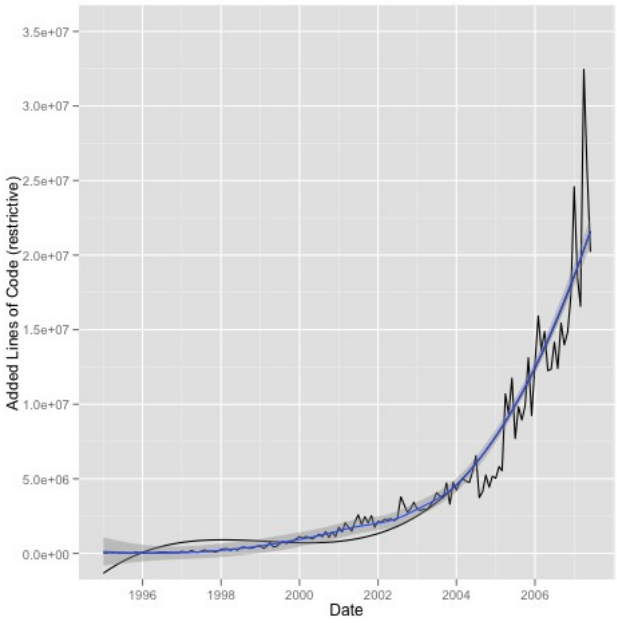

Illustration 19: Cubic model against overall added SLoC (permissive)

The model captures the trend well for both datasets from 2004 on. An explanation for the deviation in the earlier years is the heteroscedasticity of the data.

Illustration 20 and 21 show the exponential model:

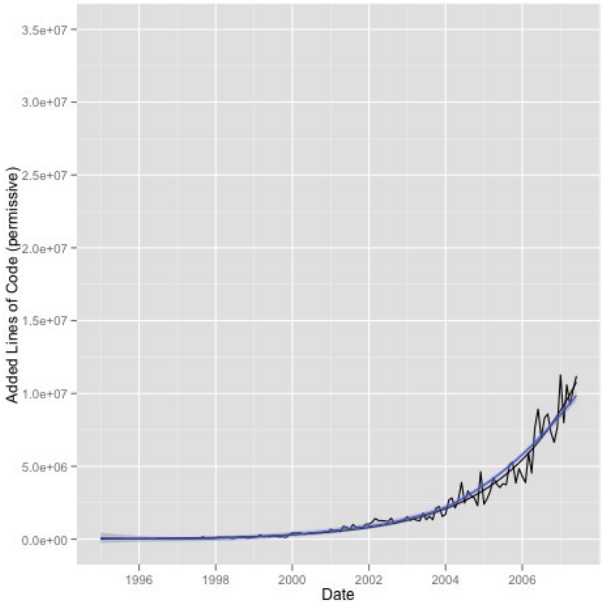

Illustration 20: Exponential model against overall added SLoC (permissive)

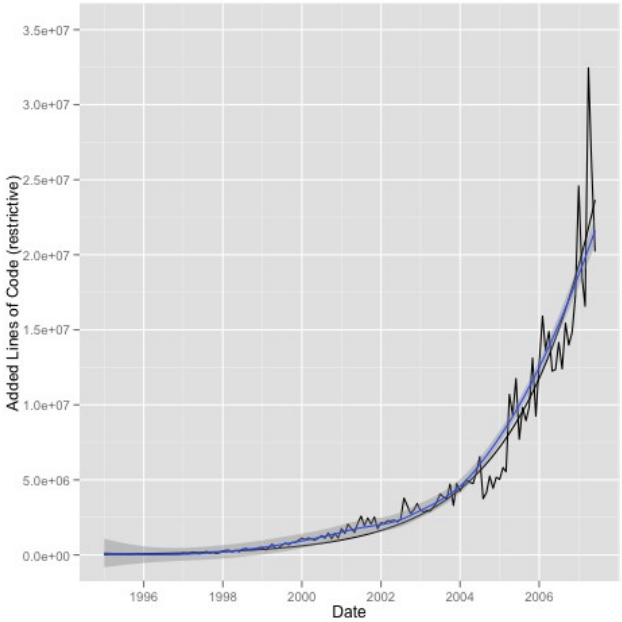

Illustration 21: Exponential model against overall added SLoC (restrictive)

The exponential model shows the best approximation of the trend.

5.3 A closer look at the exponential model

Visually, the exponential model explains the trend best for both the restrictive and permissive sets. Around 2002 there is a period of underfitting. That period shows up more drastically in the residual plots (Illustration 22 and 23):

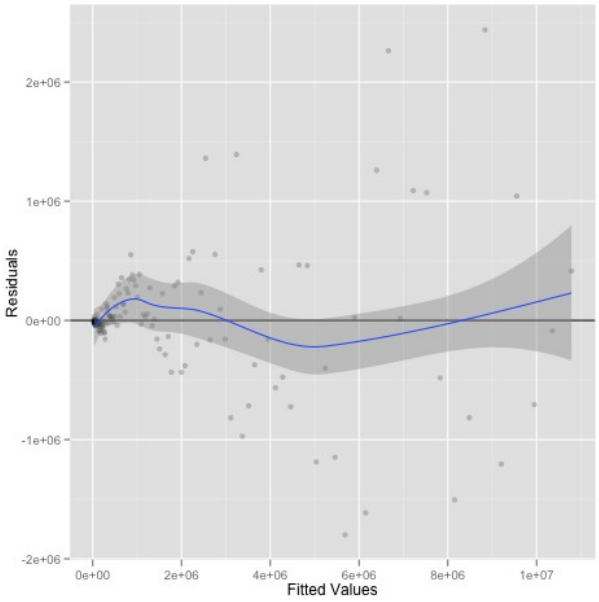

Illustration 22: Fitted values of exponential model against residuals with Loess curve in blue (permissive)

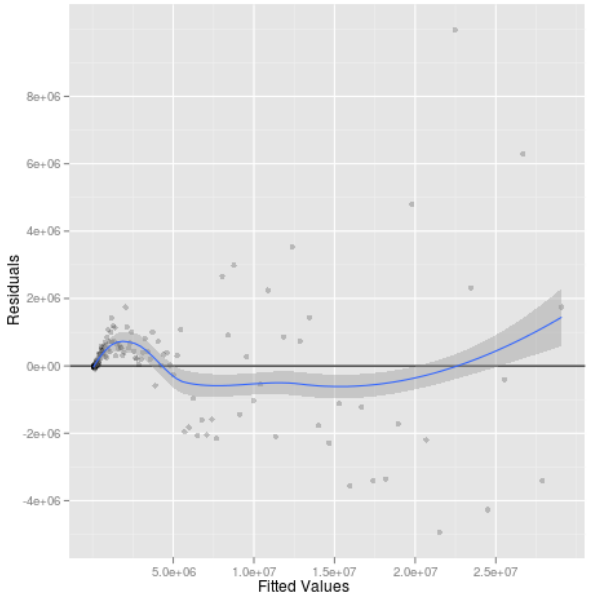

Illustration 23: Fitted values of exponential model against residuals with Loess curve in blue (restrictive)

The plots both suggest additional structure in the data and variance heterogeneity. A plot of the absolute residuals against the fitted values gives certainty (Illustration 24 and 25):

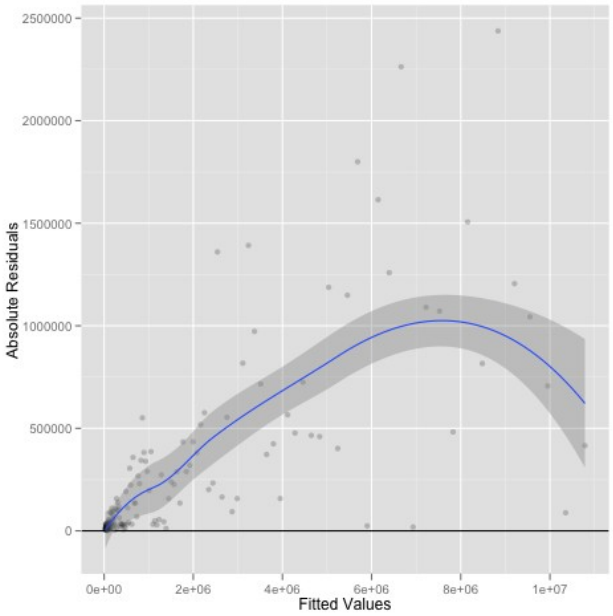

Illustration 24: Fitted values of exponential model against absolute residuals with Loess curve in blue (permissive)

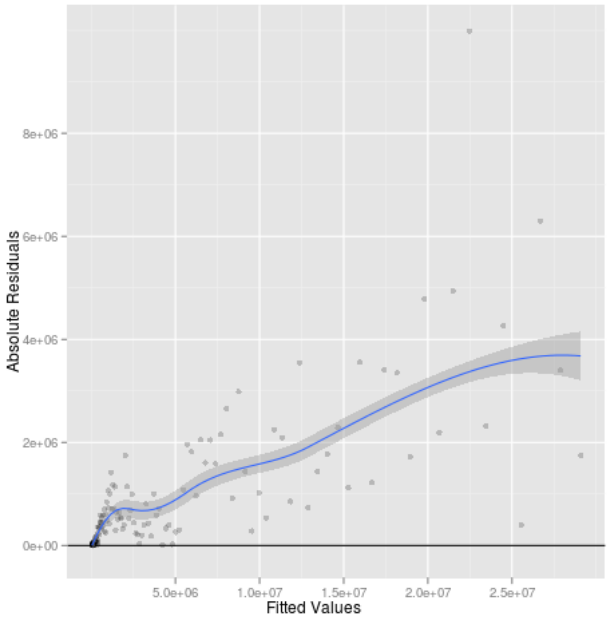

Illustration 25: Fitted values of exponential model against residuals with Loess curve in blue (restrictive)

The variance of the absolute residuals is increasing with increasing fitted values, which indicates variance heterogeneity[22]. There is also a first impression of the underlying additional structure as the increase of the variance seems to be taking place in two segments. One can roughly fit two lines with different slopes, a short one for the lower fitted values and a longer one for the larger values. Note that the Loess curve roughly behaves like two straight lines, which shows that the error is multiplicative and not additive as required by the model[23].

The QQ-Plots shows that the residuals are not normally distributed, but form a distribution that can be linearly transformed to a normal distribution (Illustration 26 and 27):

22 See Ritz and Streibig (2008) [37] p. 80 for details.

23 A formal test on heteroscedasticity was not conducted because it can be seen from the plot of the absolute residuals.

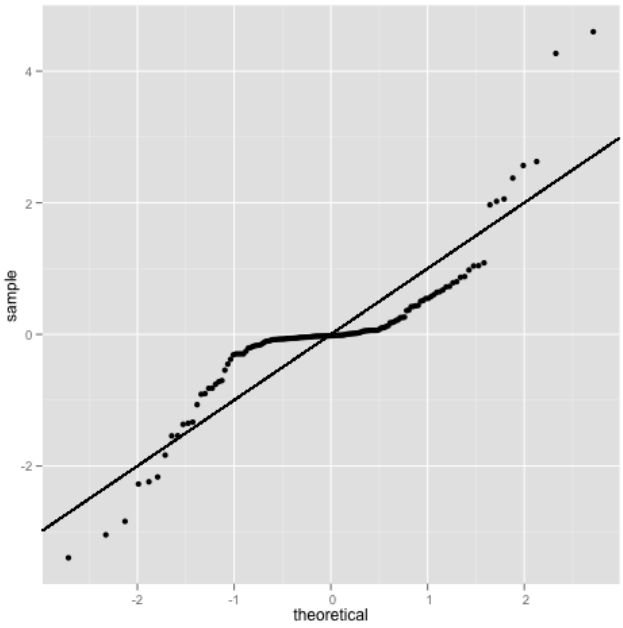

Illustration 26: QQ-plot exponential model (permissive)

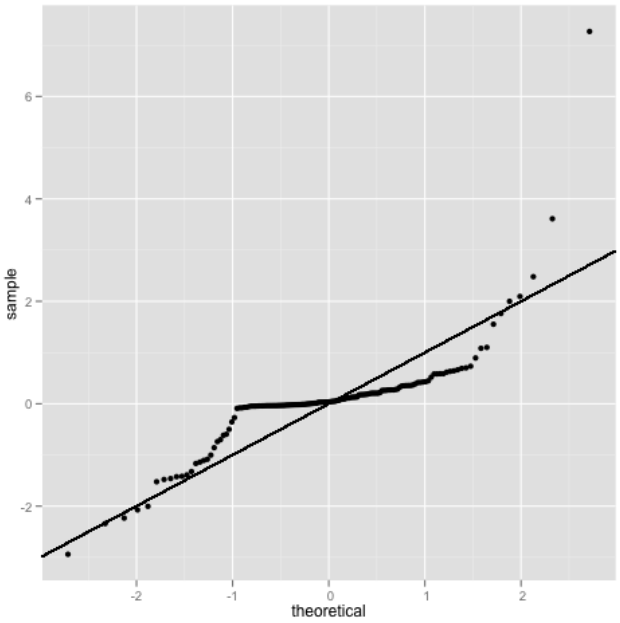

Illustration 27: QQ-plot exponential model (restrictive)

5.4 Transforming the response

As a remedy for this model violation, a logarithmic transform of the response (added SLoC) and a successive linear regression was conducted,[24] transforming the exponential model $y \sim y0 * \exp(a * x)$ to the linear model $\ln(y) \sim \ln(y0) + a * x$. This way the multiplicative error becomes and additive one: $\ln(y * \varepsilon) = \ln(y) + \ln(\varepsilon)$.

The logarithmic transform of the response yielded the following plot, further indicating that the model might need to be split into two periods (Illustration 28 and 29):

24 See Fahrmeir (2009) p. 71 for details [38]

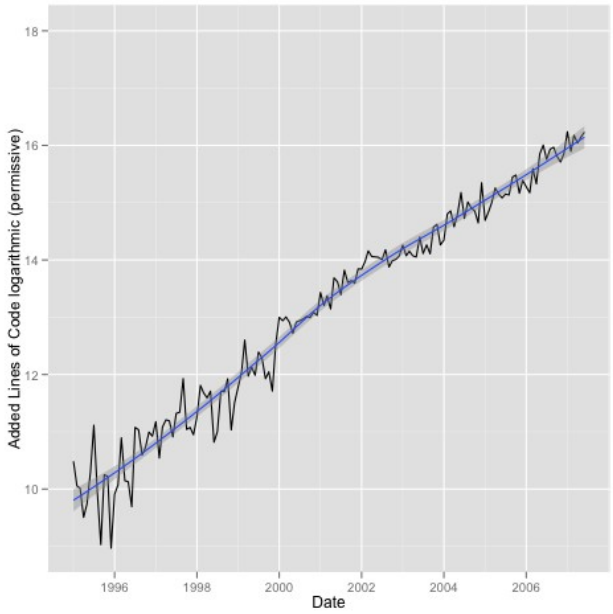

Illustration 28: Logarithmic added SLoC with Loess curve in blue (permissive)

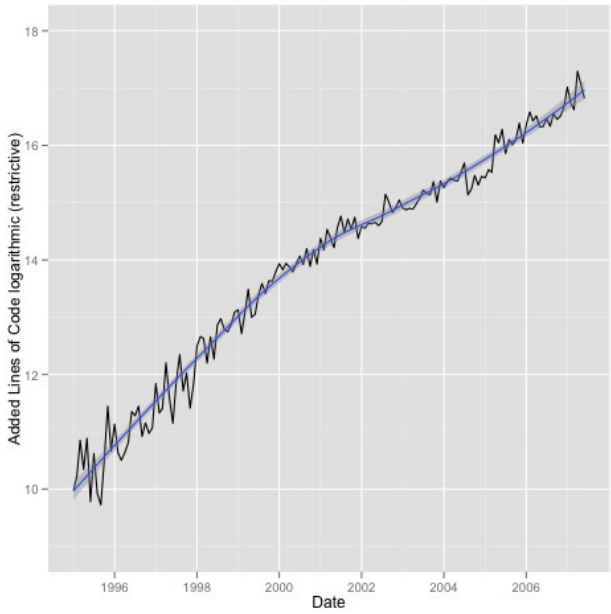

Illustration 29: Logarithmic added SLoC with Loess curve in blue (restrictive)

5.4.1 Linear Regression on the log-transformed response

A preliminary linear regression was conducted on both sets, yielding models with an adjusted Pearson's r^2 [25] of 0.9679551 for the permissive set and 0.9604772 for the restrictive (Illustration 30 and 31).

25 The adjusted Pearson's r^2 (obtained from the model fit by summary(model)$adj.r.squared in R) needs to be computed on the logarithmic data in this case since the heteroscedasticity of the non-transformed data introduces a bias towards the larger values. The adjusted version is used to be able to compare the model later on to a segmented linear model which has more degrees of freedom. The values can **not** be used to compare the linear models on log-transformed response to the normal exponential model on the normal response.

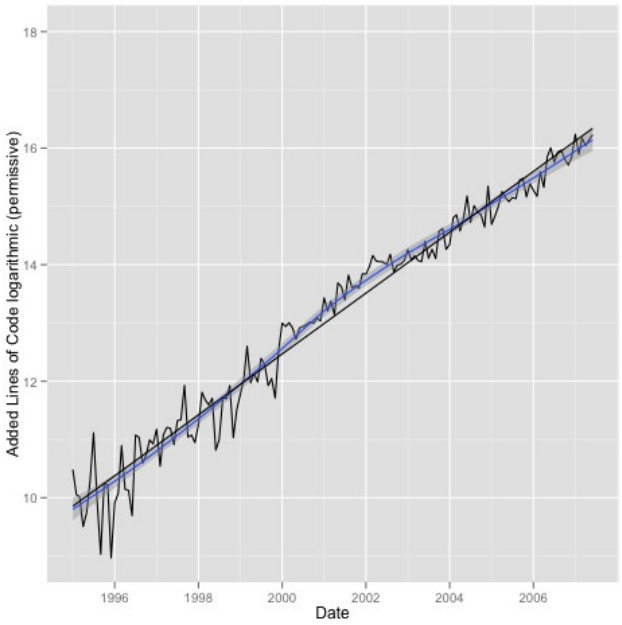

Illustration 30: Linear model against logarithmic added SLoC (permissive)

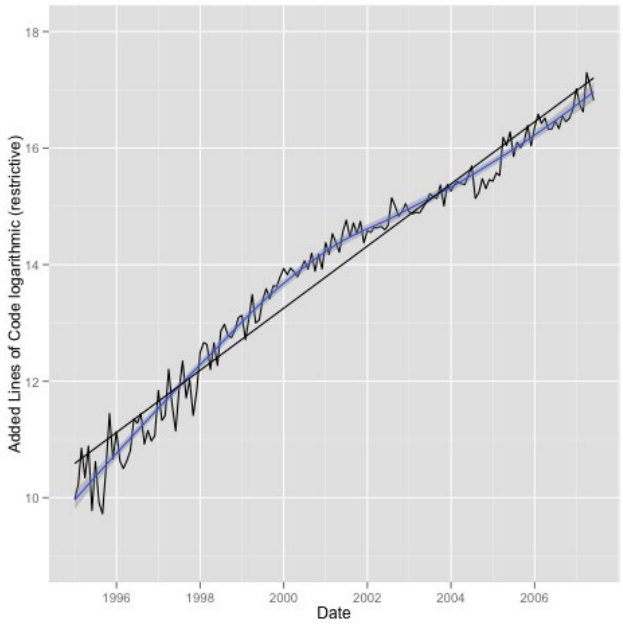

Illustration 31: Linear model against logarithmic added SLoC (restrictive)

The plot of the residuals showed a distribution without heteroscadasticity[26] and also made the underlying structure a lot more visible (Illustration 32 and 33):

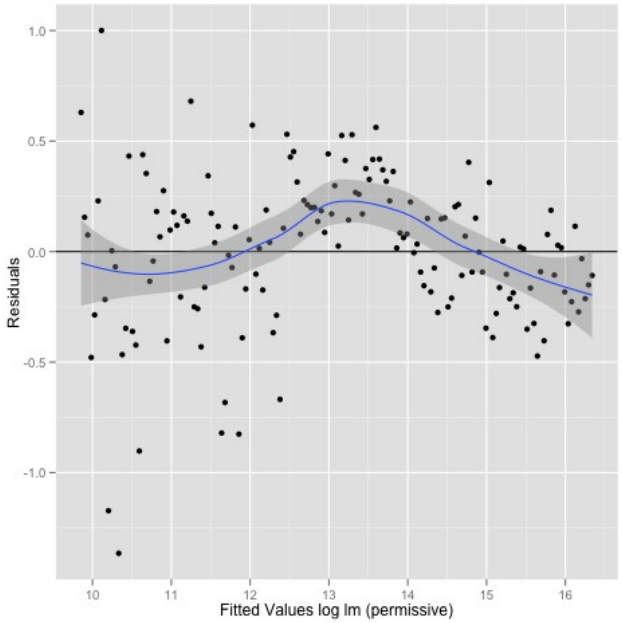

Illustration 32: Fitted values of linear model on logarithmic data against residuals with Loess curve in blue (permissive)

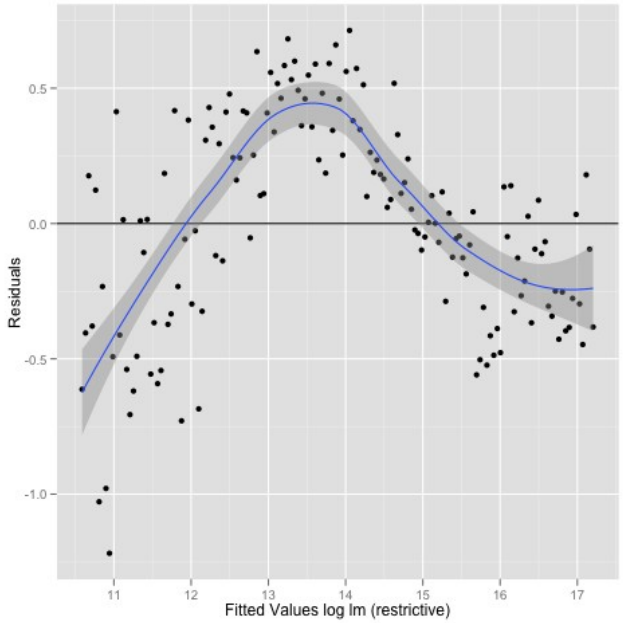

Illustration 33: Fitted values of linear model on logarithmic data against residuals with Loess curve in blue (restrictive)

The QQ-plot showed that the residuals where now nearly normally distributed yet it also showed different slopes for smaller and larger values (Illustration 34 and 35):

26 No heteroscedasticity in sense that the errors are no longer multiplicative but additive. The additional structure is another kind of heteroscedasticity which needs to be addressed.

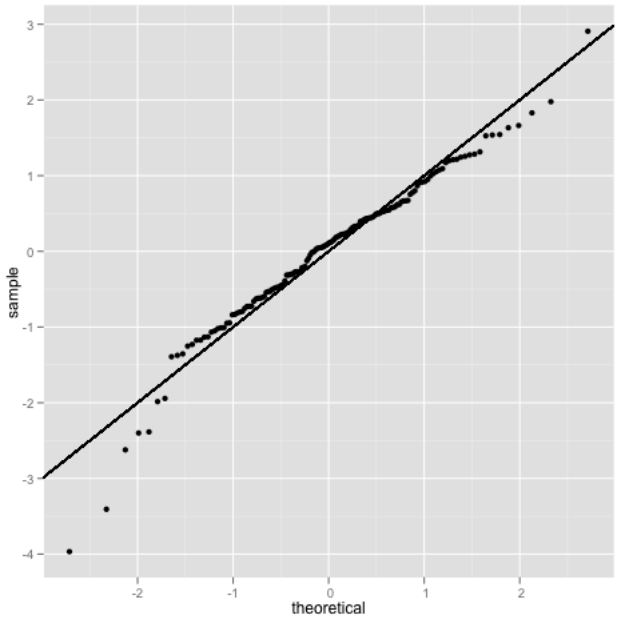

Illustration 34: QQ-plot linear model on log-transformed response (permissive)

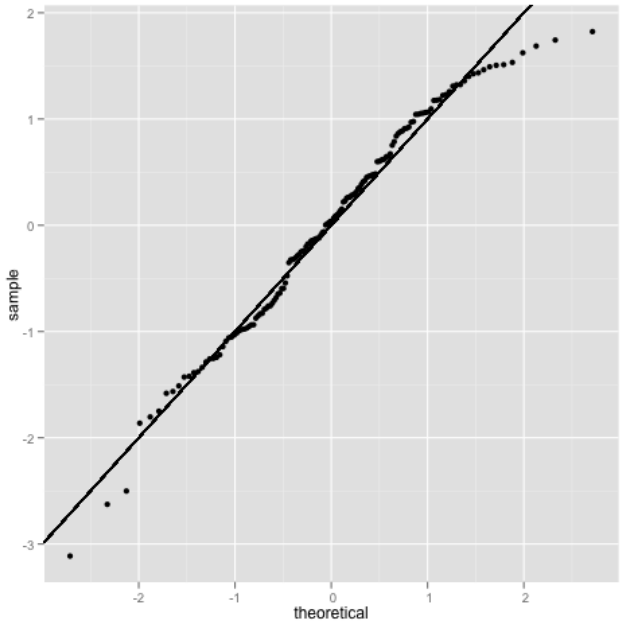

Illustration 35: QQ-plot linear model on log-transformed response (restrictive)

5.4.2 Segmenting the linear model

5.4.2.1 Ordinary Least-Squares Approach

To further adapt the linear model, a segmented regression with one break-point was conducted using the R function 'segmented' from the package 'segmented' [39] (Illustration 36 and 37):

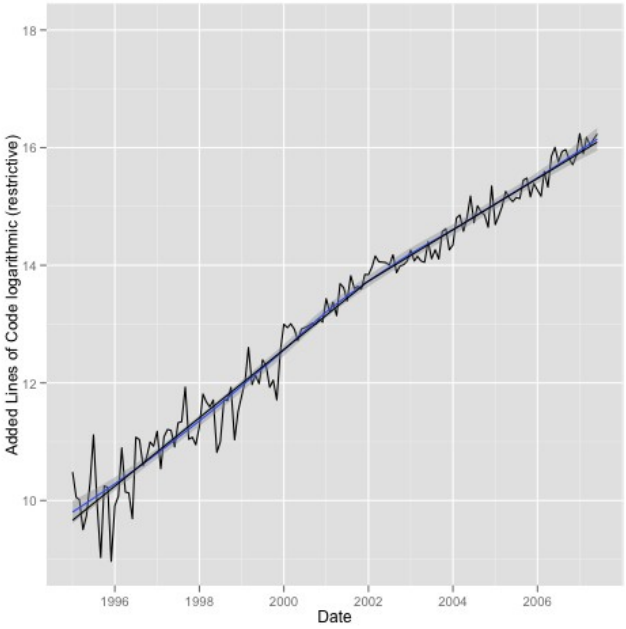

Illustration 36: Segmented linear model against logarithmic added SLoC (permissive)

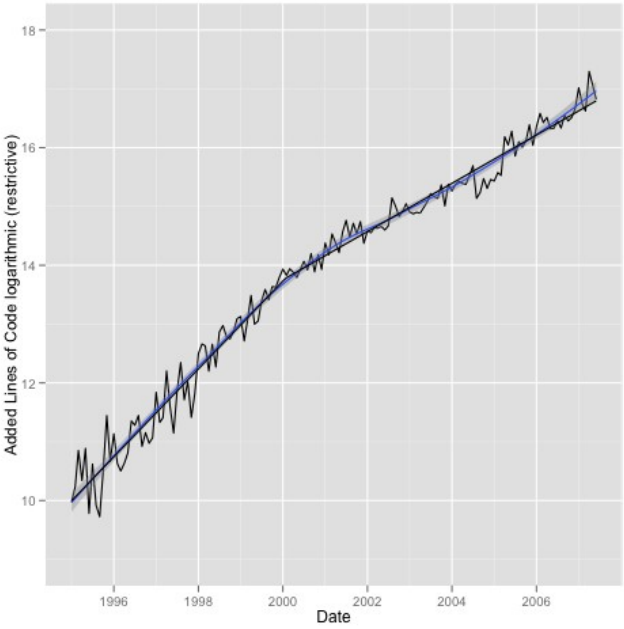

Illustration 37: Segmented linear model against logarithmic added SLoC (restrictive)

The adjusted Pearson's r^2 of the segmented model resulted in 0.9720301 for the permissive set (a change of +0.004074937 compared to the non-segmented model) and 0.9820016 for the restrictive (a change of +0.02152445).

The plot of the residuals showed that the underlying structure was reduced (Illustration 38 and 39). The segmented model satisfies the model assumptions a lot better[27].

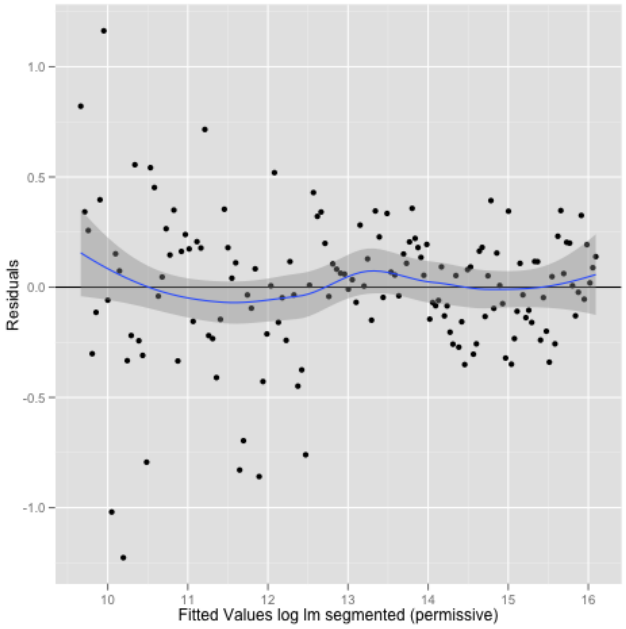

Illustration 38: Fitted values of segmented linear model on logarithmic data against residuals with Loess curve in blue (permissive)

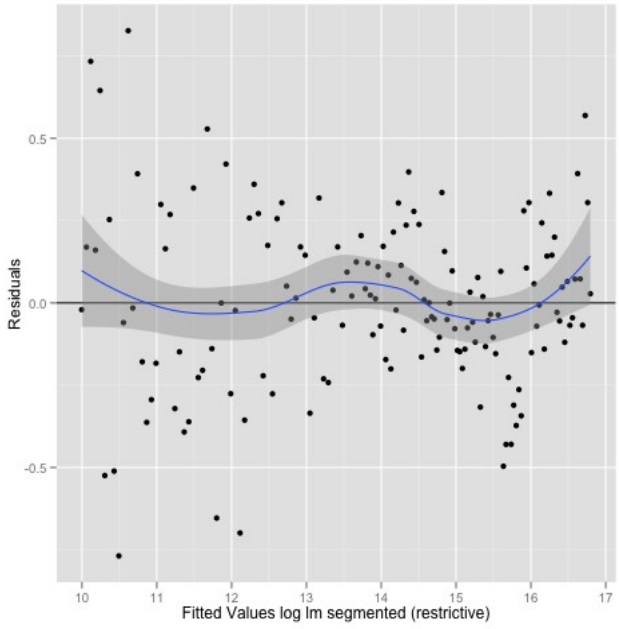

Illustration 39: Fitted values of segmented linear model on logarithmic data against residuals with Loess curve in blue (restrictive)

The QQ-plot showed a small difference between the two bins. While there was an improvement of the assumption of normal distribution of the standardized residuals for the restrictive set, the distribution of the standardized residuals of the permissive set is slightly skewed (but can still be linearly transformed to a normal distribution)[28] (Illustration 40 and 41):

27 Note that the blue Loess-curve of the residuals indicates that the change does not happen abruptly but in a smooth fashion which is natural, but not captured by the segmented model. A spline-based approach might lead to a better approximation in this case but for the sake of parsimony the segmented approach was used.

28 Which indicates that either the log-transformation or the segmentation are not an optimal approach for the permissive set. Since there is only a slight skew, the segmented model is still suitable for comparison.

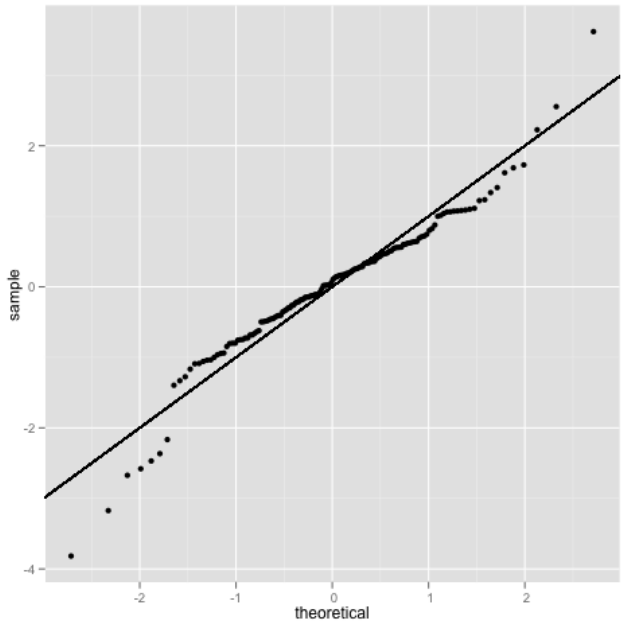

Illustration 40: QQ-plot segmented linear model on log-transformed response (permissive)

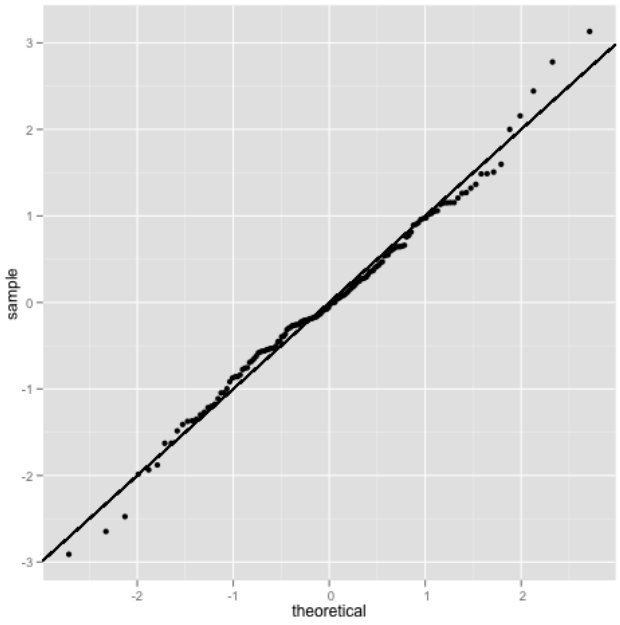

Illustration 41: QQ-plot segmented linear model on log-transformed response (restrictive)

This indicates that either the log-transformation or the segmentation are not an optimal approach for the permissive set. Consider that the segmentation did not improve the Pearson's r^2 for the permissive model as much as for the restrictive one. Also, the estimate of the 'difference in slope' of the permissive model is less than half of the restrictive one (-3.674e-04 vs. -8.910e-04). Also see Illustration 51 and 52 where the 'difference in slope' can be read from consecutive middle-lines in the bars. Same for the absolute t value of the estimates (4.755 vs. 12.255). Both indicate that the segmented approach is less suitable for the permissive than for the restrictive set. The estimated break-points and the gap variable are robust for both sets, though[29].
Since the distribution of the Residuals of the permissive segmented linear model on log-transformed response is just slightly skewed from the assumed normal distribution, it can still be considered good enough for comparison of the models.

The linear regression used for the model makes certain assumptions regarding the residuals. The above plots show that the variance is constant (no heteroscedasticity) and that the errors are normally distributed. A third requirement is that the errors are uncorrelated. While correlation in the error is directly visible for the non-segmented approach (the clear structure in the residual plots), it's not so clear for the segmented approach. For example if a positive residual is likely to follow a positive residual and a negative residual likely to follow a negative one, this indicates a positive autocorrelation. In this case the least-squares-estimator is no longer a maximum-likelihood-estimator for the regression coefficients and another method should be used. To test for autocorrelation in the residuals, the Durbin-Watson-Test for a maximum lag of 3 was conducted (Table 3):

29 See Muggeo (2008) p. 4 for details.

License-type	Lag	Autocorrelation	D-W Statistic	p-value
Permissive	1	0.19724032	1.559559	0.002
	2	-0.08570801	2.117196	0.600
	3	-0.07591252	2.093190	0.590
Restrictive	1	0.13744496	1.724994	0.062
	2	0.03757802	1.912786	0.492
	3	-0.02042246	1.944056	0.670

Table 3: Autocorrelation and Durbin-Watson-Statistic for the segmented linear models up to lag 3

The test[30] indicates that there is some slight positive, but significant[31] autocorrelation at lag 1 for both segmented linear models[32]. But are the autocorrelation-values at lag one of about 0.2 for the permissive and 0.14 for the restrictive set large enough to be considered for the estimation of the regression coefficients? Illustration 43 and 42 show the correlogram of the studentized residuals of the segmented linear models[33]:

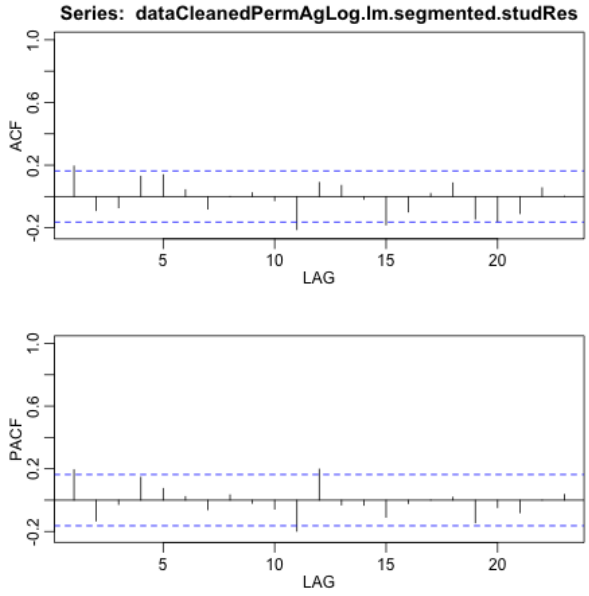

Illustration 43: Correlogram of the studentized residuals of the segmented linear model (permissive)

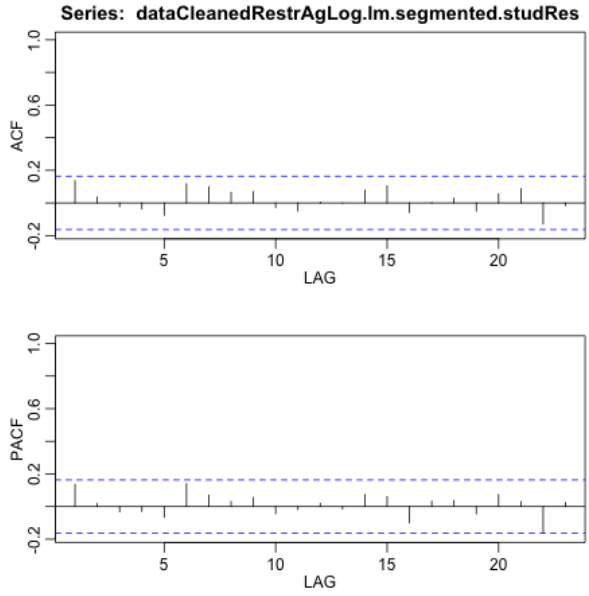

Illustration 42: Correlogram of the studentized residuals of the segmented linear model (restrictive)

For the permissive set the correlation at lag one is slightly above the suggested threshold while it stays below in the restrictive set. A slight amount of correlation can be expected from the segmented approach since it assumes an abrupt change which is unlikely in most 'natural' conditions (look at the two 'dents' in the blue Loess curve in the residual plots for both sets shown in Illustration 38 and 39).

30 For further information on the Darbin-Watson-Test consult Fahrmeier (2009) p. 141

31 The Darbin-Watson-Statistic is about 2 for no autocorrelation. Values up to 0 or 4 indicate positive or negative autocorrelation.

32 The p-value for the restrictive set is slightly above the 0.05-mark, though.

33 The plots start at lag 1 and not at lag 0 like the normal plots from the acf()-function in R. The correlation at lag one is always one, thus it does not provide additional information and was omitted.

5.4.2.2 Generalized Least-Squares approach

To take the autocorrelation into account, the segmented models where re-fit[34] using the generalized least-squares (GLS) estimator which works as a maximum-likelihood-estimator even under the presence of correlation. Since the package 'segmented', which was used for the segmentation of the linear model, does not work with GLS, the segmentation had to be created manually using the estimated break-points from the segmented models[35]. To see whether the fit improves under the assumption of correlation, the algorithm was run twice, once with and once without provision for correlation. The comparison of model-selection criteria is shown in Table 4:

License-type	Correlation	AIC	BIC	logLik
Permissive	No	136.5919	148.5537	-64.29597
	AR(1)	129.4288	144.3809	-59.71439
Restrictive	No	73.63248	85.59421	-32.81624
	AR(1)	71.90298	86.85514	-30.95149

Table 4: Comparison of model-selection criteria for the generalized least-squares fit with and without provision of correlation (segmented approach).

AIC (Aikake Information Criterion) and absolute Log-likelihood improved for for models, the BIC (Bayesian Information Criterion) on the restrictive set got worse when correlation is taken into account. Furthermore the difference is rather slight for the restrictive set. For the restrictive set, the AIC and logLik-results where favored over the BIC since the the sample was taken from a dataset that roughly covers 30% of all open source projects at the time[36] and AIC favors models that fit the current data[37].

The resulting fit is just minimally different from the linear approach with ordinary least-squares estimation (Illustration 44 and 45):

34 The linear model without segmentation was re-fit with GLS only for the purpose of a Likelihood-ration-test. Due to the clear structure in the residuals of the non-segmented approach which indicates a model violation the non-segmented model was not further analyzed.

35 Note that this results in a model with one degree of freedom less (the psi-value) than the segmented linear model.

36 See Chapter 4.1 for details.

37 For a detailed comparison of AIC and BIC see Burnham (2002) [41] Chapter 6.4

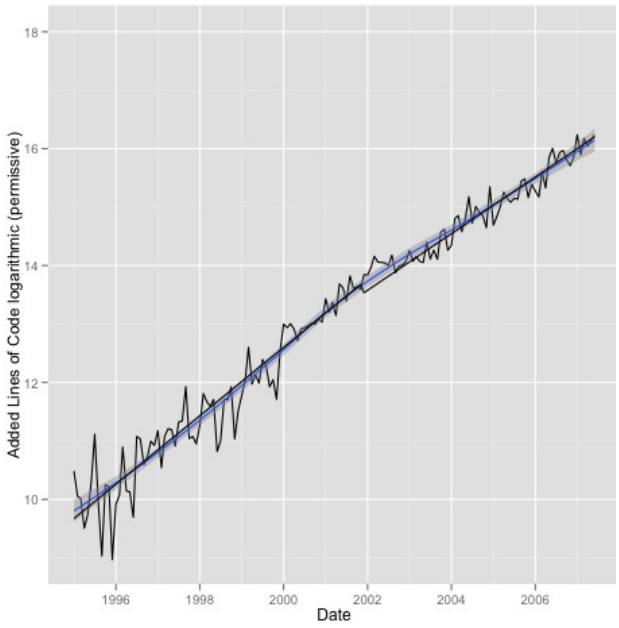

Illustration 44: Segmented linear model against logarithmic added SLoC using GLS (permissive)

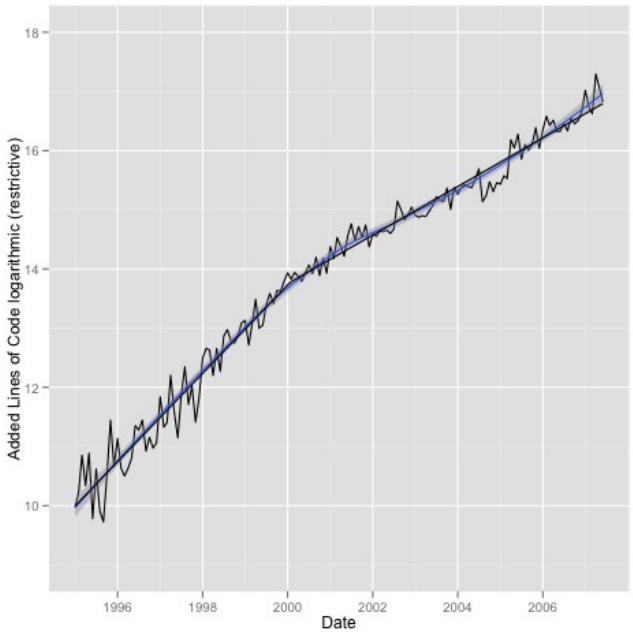

Illustration 45: Segmented linear model against logarithmic added SLoC using GLS (restrictive)

A small gap can be noticed at the break-point for the permissive set[38] while for the restrictive one there is hardly any difference to the normal linear segmented approach. Illustration 46 and 47 show the residuals:

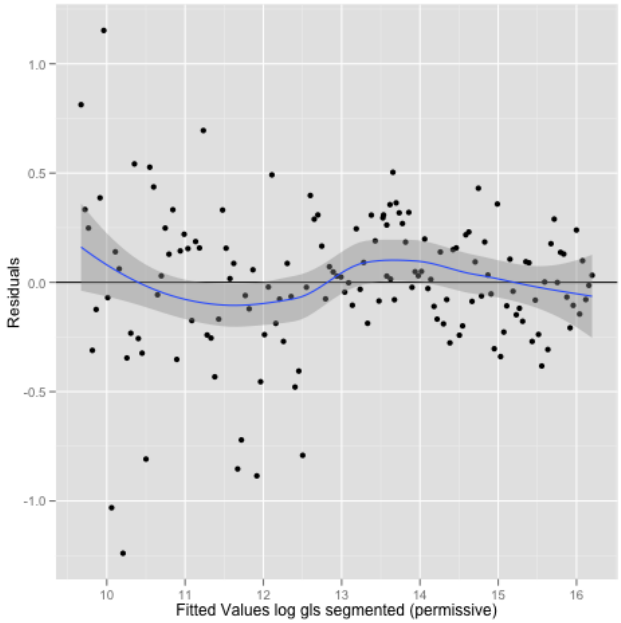

Illustration 46: Fitted values of segmented linear model using GLS on logarithmic data against residuals with Loess-curve in blue (permissive)

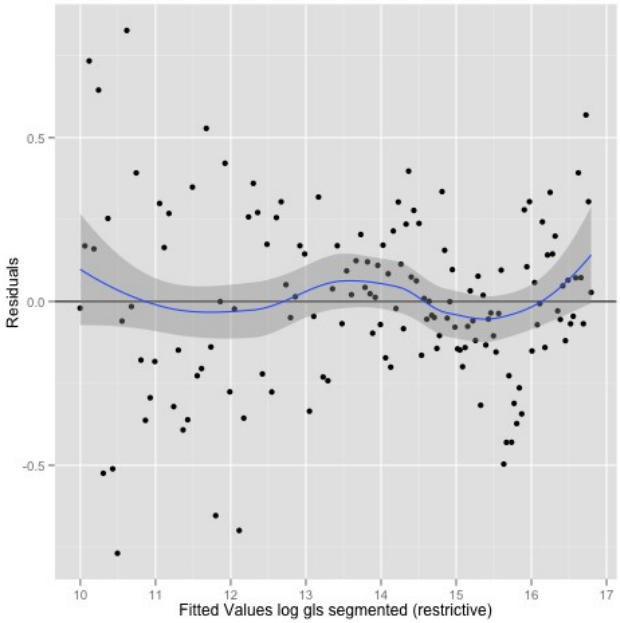

Illustration 47: Fitted values of segmented linear model using GLS on logarithmic data against residuals with Loess-curve in blue (restrictive)

38 Which indicates that the estimate for the break-point that was taken from the segmented linear model is not perfect for the GLS approach.

The residuals show some considerable difference for the permissive set where the blue Loess-curve lost the second 'dent'. The restrictive set's residuals are just arranged a little more compact. Illustration 48 and 49 show the QQ-plots:

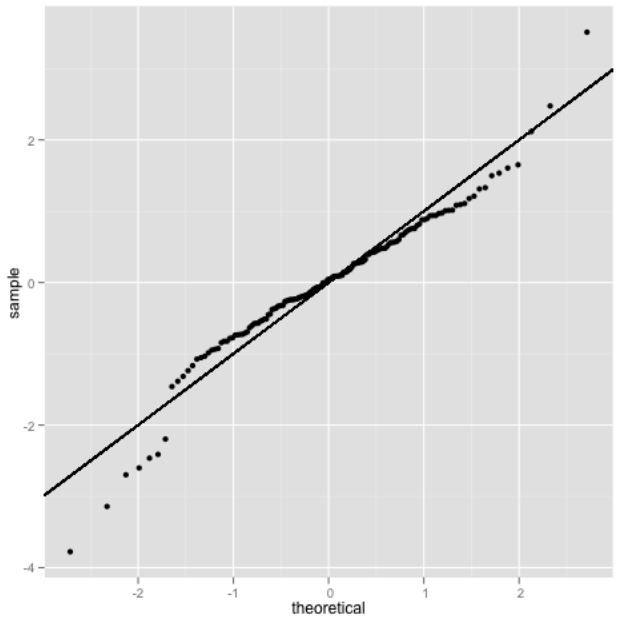

Illustration 48: QQ-plot segmented linear GLS model on log-transformed response (permissive)

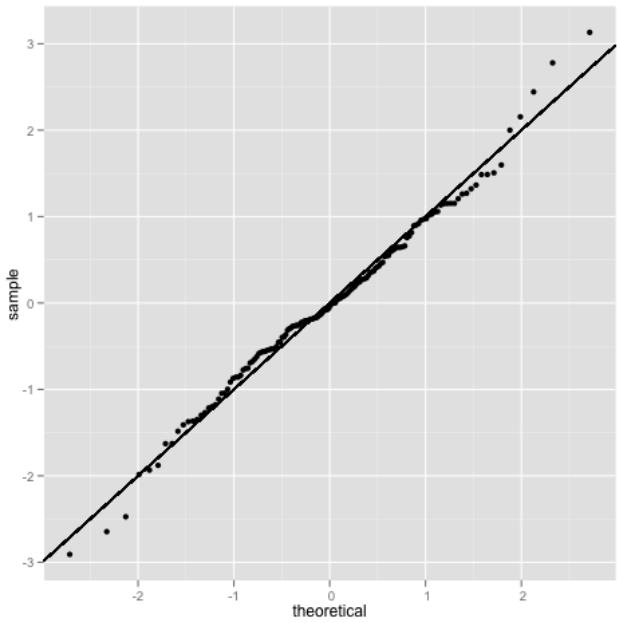

Illustration 49: QQ-plot segmented linear GLS model on log-transformed response (restrictive)

Especially in the permissive set, the distribution is way less distorted now. Yet the distribution of the permissive set is still slightly skewed from a normal distribution, though closer.

5.5 Discussion of the models

5.5.1 Non-segmented linear approach

The non-segmented approach on the log-transformed response yielded a linear model with an intercept of -3.203185072 and a slope of 0.001430132 for the permissive set and an intercept of -2.735973871 and a slope of 0.001459238 for the restrictive one. The estimated parameters of the slope[39] don't overlap within a confidence interval of 45%[40]. That means with a confidence of 45% the total growth of open source Software from 1995 to 2007 differs in terms of license-type, with the restrictive set showing a slightly faster growth. By increasing the confidence level, the upper bound of the permissive slope starts to overlap with the lower bound of the permissive slope[41]. At 95% confidence, the estimates of the slope of one model are inside the confidence interval of the other one. A comparison is shown in Table 5.

39 Which is the parameter that determines growth.

40 Used: a level of 0.45 in the R-function confint.lm().

41 Note that this means that there is a 27,5% chance that the slope of the permissive model exceeds the upper bound of the confidence interval and a 27,5% chance that the slope of the restrictive model is less steep than the lower bound of the confidence interval. Thus the overall chance that the restrictive set does **not** grow faster than the permissive one is **at most** 55% according to the model.

License-type	Model on log-transformed response	45% conf. int. of slope		95% conf. int. of slope	
		27,5%	72,5%	2,5%	97,5%
Permissive	y = -3.203185072 + 0.001430132x	0.001417361	0.001442902	0.001388011	0.001472253
Restrictive	y = -2.735973871 + 0.001459238x	0.001444711	0.001473765	0.001411323	0.001507152

Table 5: Comparison of non-segmented linear models on log-transformed response for the restrictive and permissive set and confidence intervals for the slope.

The following plot shows the estimates of the slopes of the restrictive model and the permissive model with 95% confidence intervals, which are clearly overlapping (Illustration 50):

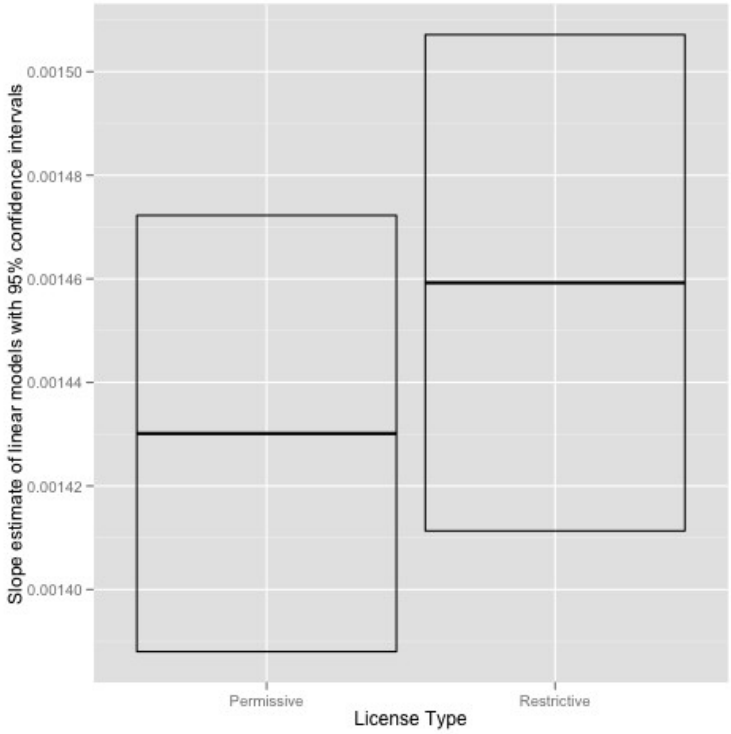

Illustration 50: Slope estimates of the linear models on log-transformed response with 95% confidence intervals.

5.5.2 Segmented linear approach

The segmented approach yielded models with one break-point each. The intercept was estimated as -4.860492 for the permissive set and -8.677363 for the restrictive one. The break-points where estimated around 2001-12 for the permissive set and 2000-02 for the restrictive. The slopes for the first periods where estimated as 0.001591 for the permissive and 0.002045 for the restrictive set with no overlapping confidence interval at a level of 95%. For the second periods, the slope-estimates where 0.001195 for the permissive set and 0.001123 for the restrictive with no overlap at a confidence level of 51%.
This means that while in the first period the total growth of the restrictive set is significantly higher than the permissive one, things changed in the second period with the permissive set showing

higher total growth. The confidence bands show, that for the second period, the difference in growth is **not significant.** A comparison of the slopes is shown in Table 6:

License-type	Period	Slope on log(response)	51% conf. int. of slope		95% conf. int. of slope	
			25.5%	75.5%	2,5%	97,5%
Permissive	1	0.001591	0.001557	0.001624	0.001495	0.001686
	2	0.001195	0.001149	0.001241	0.001063	0.001327
Restrictive	1	0.002045	0.002001	0.002089	0.001920	0.002170
	2	0.001123	0.001099	0.001148	0.001053	0.001194

Table 6: Comparison of the slope of the segmented linear models on log-transformed response for the restrictive and permissive set including confidence intervals.

The following two plots show the estimates of the slopes of the segmented linear restrictive and permissive model with 95% confidence intervals, on the left the first, and on the right the second period. While in the first period the bands don't overlap, in the second there is a considerable overlap (Illustration 51 and 52):

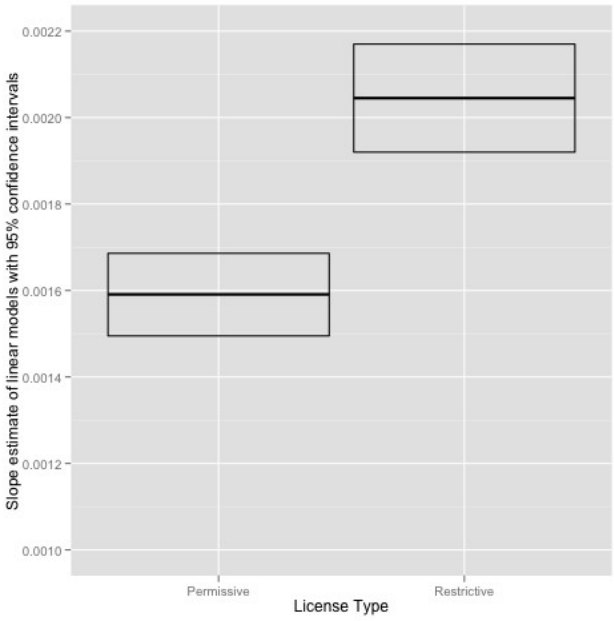

Illustration 51: Slope estimates of the first period of the segmented linear models on log-transformed response with 95% confidence intervals.

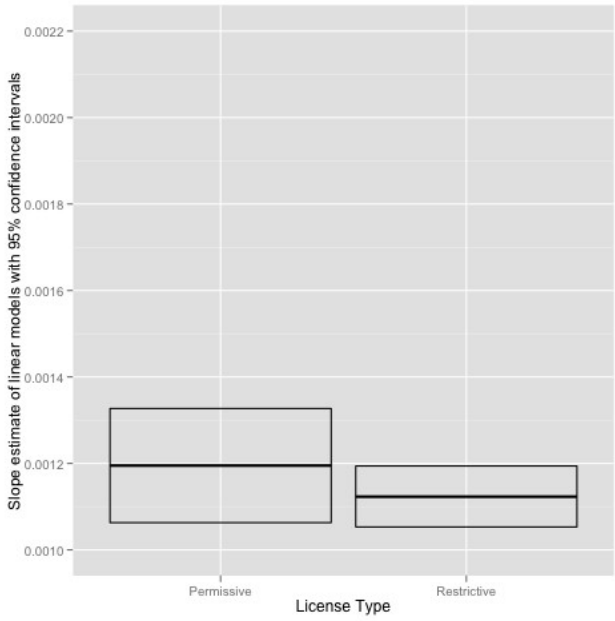

Illustration 52: Slope estimates of the second period of the segmented linear models on log-transformed response with 95% confidence intervals.

As already indicated in Chapter 5.4.2, the difference-in-slopes for the two restrictive periods is more than two times higher than in the restrictive set.

The 95% confidence intervals for the break-points are shown in Table 7:

License-type	Estimated break-points (rounded to months)	95% conf. int. of break-points	
		2,5%	97,5%
Permissive	2001-12	2000-06	2003-05
Restrictive	2000-02	1999-08	2000-08

Table 7: Estimated break-points for the segmented linear model on log-transformed response and 95% confidence intervals.

The estimated break-points with 95% confidence intervals alongside the segmented models and the log-transformed data are shown in the following plot with red representing the restrictive set and model and blue representing the permissive (Illustration 53)[42]:

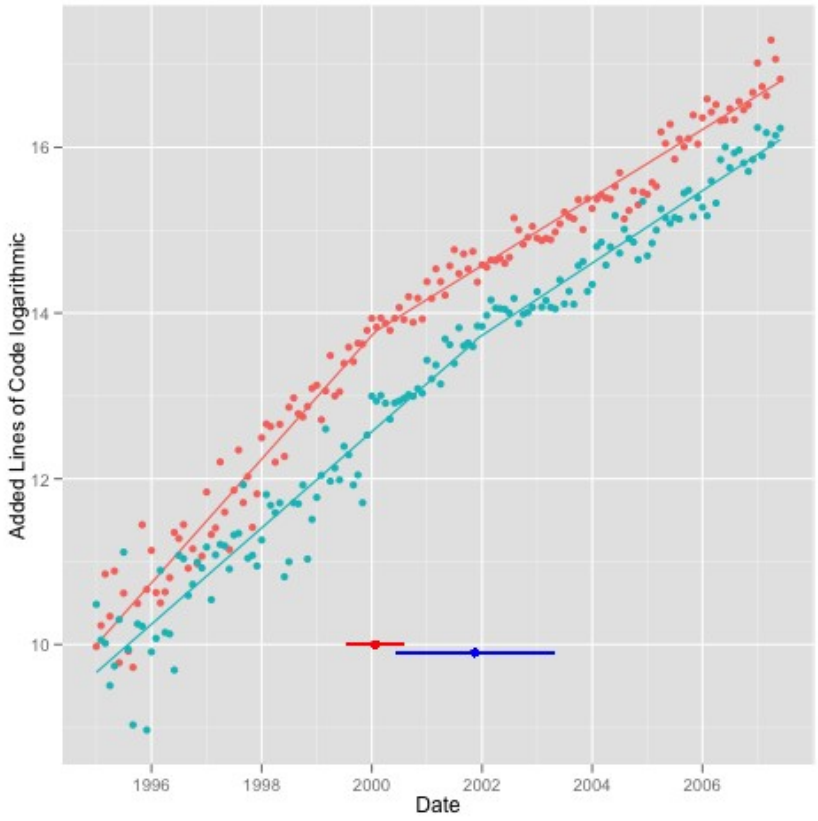

Illustration 53: Estimates of the break-points of the segmented models alongside the models and the data on log transformed response. Red indicates the restrictive set and blue the permissive.

As a formal test to compare the fit of the linear models against the segmented linear models a likelihood-ratio-test was conducted which resulted in a p-value of 0.00000008 for the permissive set and < 0.00000001 for the restrictive. Thus the non-segmented approach was rejected.

42 The larger confidence interval for the permissive model corresponds with the lower certainty of segmentation (see 5.4.2)

5.5.3 Segmented Linear Approach with Generalized Least-Squares

The intercept for the segmented linear models fitted by GLS was estimated as -4.9705448055 for the permissive set and -8.6781984894 for the restrictive one. The slope of the first segmented resulted in an estimate of 0.0016035989 for the permissive and 0.0020451526 for the restrictive set with no overlap at 95% confidence intervals. The slope-estimates for the second segments resulted in 0.001326682 for the permissive set and 0.001123542 for the restrictive with no overlap of confidence intervals up to 75%. This means that after taking correlation into account, the confidence intervals still show no significant evidence that the permissive set grows faster than the restrictive, but the chances are no longer 51:49 but rather 75:25. A comparison of slopes is shown in Table 8:

License-type	Period	Slope on log(response)	75% conf. int. of slope		95% conf. int. of slope	
			12.5%	87.5%	2.5%	97.5%
Permissive	1	0.0016035989	0.0015325407	0.0016746570	0.001482530	0.0017246677
	2	0.001326682	0.001225261	0.001428103	0.001153881	0.001499483
Restrictive	1	0.0020451526	0.001982252	0.0021080535	0.001937982	0.0021523232
	2	0.001123542	0.001030812	0.001216273	0.000965548	0.001281537

Table 8: Comparison of the slope of the segmented linear models using GLS on log-transformed response for the restrictive and permissive set including confidence intervals.

The following two plots show the estimates of the slopes of the GLS-approach of the segmented linear restrictive and permissive models in the same way as done with the segmented linear approach. Once again while in the first period the bands don't overlap. In the second there is a considerable overlap, but the estimated values for the slopes are outside the confidence intervals of the other set now (Illustration 54 and 55):

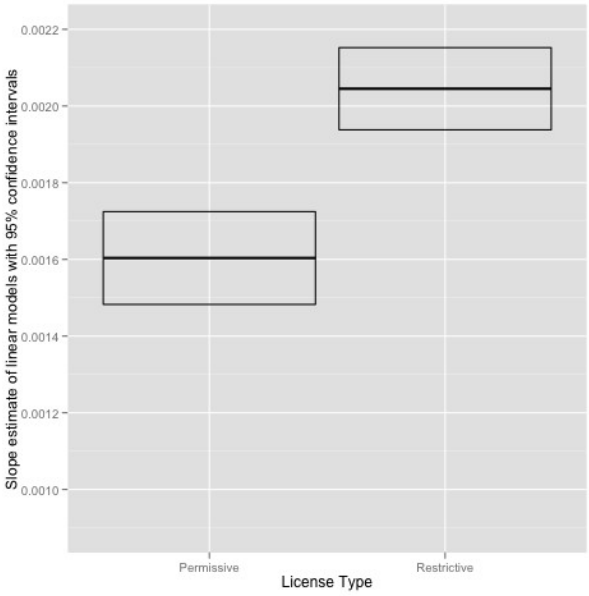

Illustration 54: Slope estimates of the first period of the segmented linear models fit by GLS on log-transformed response with 95% confidence intervals.

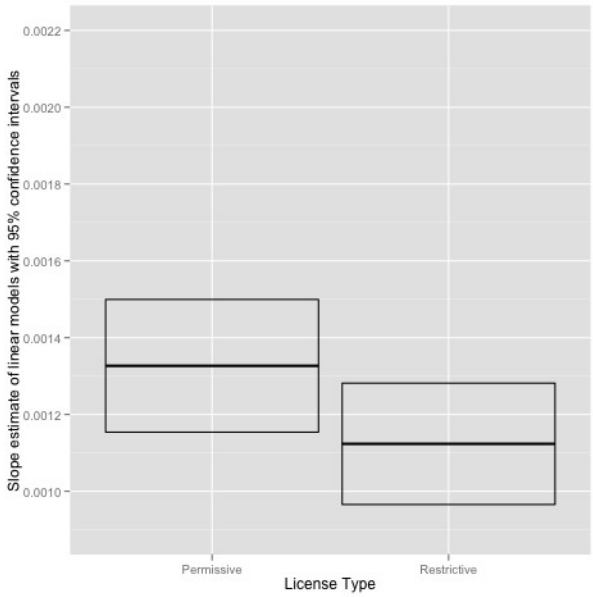

Illustration 55: Slope estimates of the second period of the segmented linear models fit by GLS on log-transformed response with 95% confidence intervals.

Even though the confidence intervals for the second period got broader, the overlap has been reduced. There are no confidence intervals for the estimates of the the break-points since the break-point estimates from the segmented models have been used[43].

A Likelihood-ratio-test was conducted again, this time against a linear fit with GLS without correlation specified. The test resulted in a p-value of 0.002487426[44] for the permissive set and < 0.00000001 for the restrictive. Thus the non-segmented approach was rejected also for the models using GLS.

5.5.4 Discussion of confidence

Only the difference-in-slopes for the first segment of the linear models is significant (>95% confidence). Difference-in-slopes for the normal linear approach is not significant (45% confidence) and neither is the difference-in-slopes for the second segment of the segmented linear approach (51% confidence). The difference-in-slopes for the second period of the segmented GLS-approach is also not significant, but there is an indication of difference. An overview is shown in Table 9:

Approach	Result (in total growth)	Confidence	Confidence GLS
Non-Segmented	Restrictive > Permissive	45%	-
Segmented	First Period: Restrictive > Permissive	>95%	>95%
	Second Period: Restrictive < Permissive	51%	75%

Table 9: The differences in growth binned by model and license-type with confidence intervals.

While there came no certainty from the normal approach, the segmented models show a significant difference in growth binned by license-type from 1995 to roughly 2000/2002. During that time, the restrictive projects grew in total faster than the permissive-licensed ones. Then a change happened and the growth slowed for both license-types, but the restrictive set showed a stronger slowdown than the permissive one. Since the changing-point, the growth can not be distinguished with 95% confidence, yet the results from the GLS approach indicate that the trend was reversed and the permissive set grows faster since then.

5.6 The models transformed to normal scale

Transforming the linear models back to normal scale yielded the following exponential models (Table 10):

43 Note that the estimates of the break-points might be different with GLS but no R package for this approach was available at the time of writing.

44 Due to the fact that the segmented linear models with GLS lack one degree of freedom (the break-point was taken from the normal segmented linear models, the test was also run with one theoretical additional degree of freedom which resulted in a p-value of 0.01030723 for the permissive set, which also rejects the non-segmented approach.

License-type	Approach	Model[45]
Permissive	Normal	$y = 0.04063258 * e^{0.001427541 * x} * e^{\varepsilon}$
	Segmented	$y = 0.007746672 * e^{0.001590648 * x} * e^{(-0.0003956382 * (x-\psi)_t)} * e^{\varepsilon}$
	Segmented GLS	$y = 0.006939366 * e^{0.001603599 * x} * e^{(-0.0002769172 * (x-\psi)_t)} * e^{\varepsilon}$
Restrictive	Normal	$y = 0.06483084 * e^{0.001453936 * x} * e^{\varepsilon}$
	Segmented	$y = 0.0001703999 * e^{0.002045079 * x} * e^{(-0.0009216163 * (x-\psi)_t)} * e^{\varepsilon}$
	Segmented GLS	$y = 0.000170258 * e^{0.002045153 * x} * e^{(-0.0009216103 * (x-\psi)_t)} * e^{\varepsilon}$

Table 10: Comparison of the linear models transformed to the non-logarithmic scale.

The back-transformed models include the error term, because the error roughly has a mean of zero for the linear models on the log-transformed response, which is no longer the case when the models get transformed back to normal scale. An estimate of the bias was conducted using the "smearing estimate of bias" for residuals that are not normally distributed[46]. The bias needs to be taken into account when the models are used for prediction and is listed in Table 11:

License-type	Approach	Error-bias e^{ε}
Permissive	Normal	1.056218 (5.6%)
	Segmented	1.050099 (5.0%)
	Segmented GLS	1.049469 (4.9%)
Restrictive	Normal	1.074518 (7.5%)
	Segmented	1.034837 (3.5%)
	Segmented GLS	1.034829 (3.5%)

Table 11: Error-bias of the transformed linear models

The following two plots show the non-segmented linear models transformed to normal response (Illustration 56 and 57):

45 For the segmented models, $(x-\psi)_t$ defines a function where ψ is the break-point and $(x-\psi)_t$ is 0 for $(x<\psi)$

46 See Newman (1993) for details [40]

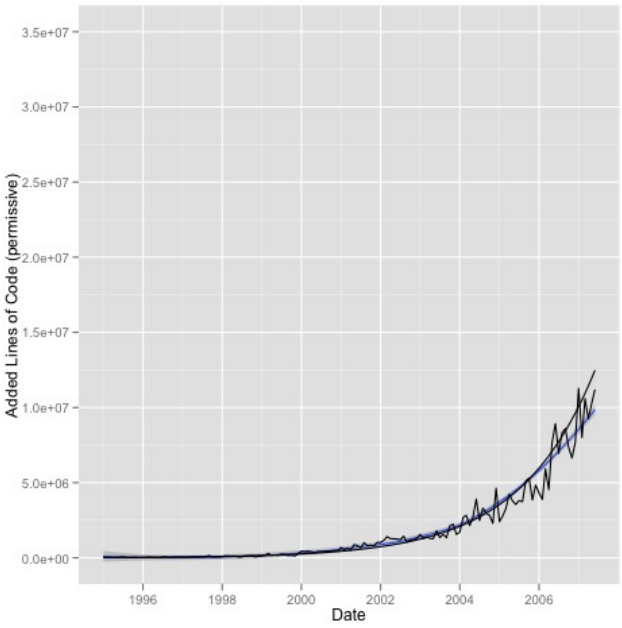

Illustration 56: Exponentialized linear model against overall added SLoC (permissive)

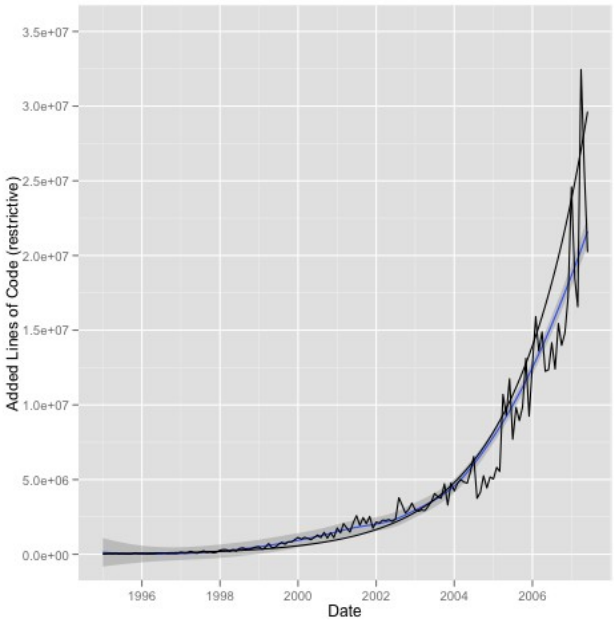

Illustration 57: Exponentialized linear model against overall added SLoC (restrictive)

The models are less accurate at the higher values toward the end of the analyzed period compared to the normal exponential model[47] as these where the values who's residuals have more weight in the normal approach.

The next two plots show the segmented linear models transformed to normal response (Illustration 58 and 59):

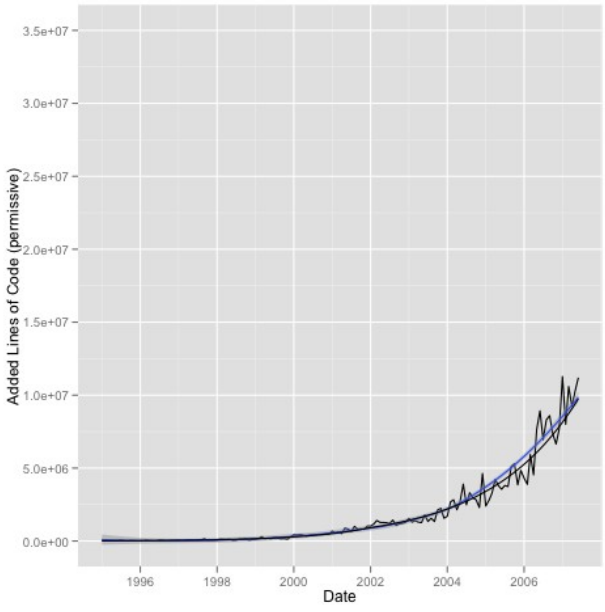

Illustration 58: Exponentialized segmented linear model against overall added SLoC (permissive)

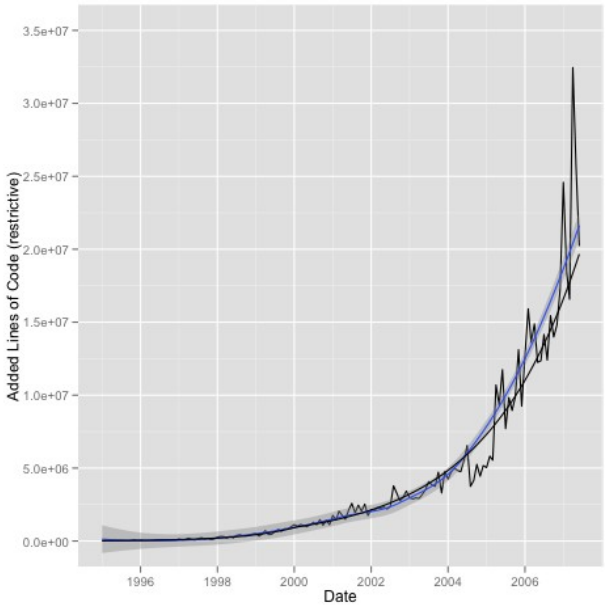

Illustration 59: Exponentialized segmented linear model against overall added SLoC (restrictive)

47 See Chapter 5.2 for comparison.

Once again the models visually deviate at the higher values of the response, an effect that gets visually intensified by the high slope in that area. A comparison with the normal exponential model still shows that the exponentialized segmented linear approach follows the characteristics of the Loess-curve better. For example in the restrictive set, the exponential model curves stays strictly below the Loess-curve for an segment ranging from roughly 1999 to the end of 2006 while in the exponentialized segmented linear approach a similar segment ranges only from the second half of 2004 until the middle of 2007.

The next two plots show the segmented linear models with GLS transformed to normal response (Illustration 60 and 61):

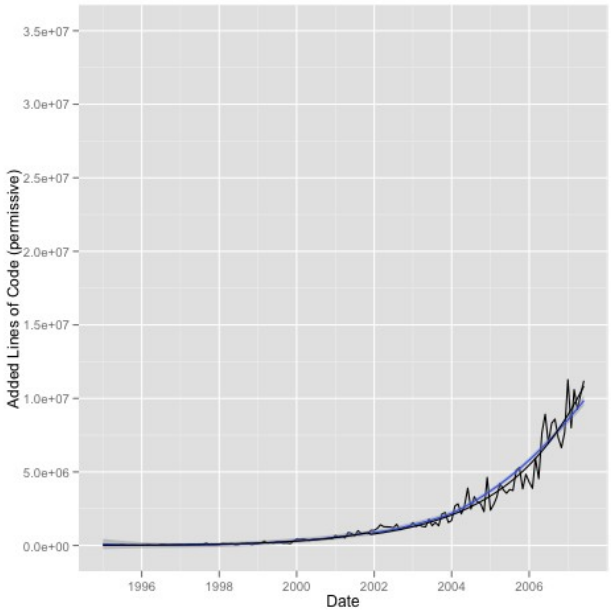

Illustration 60: Exponentialized segmented linear model using GLS against overall added SLoC (permissive)

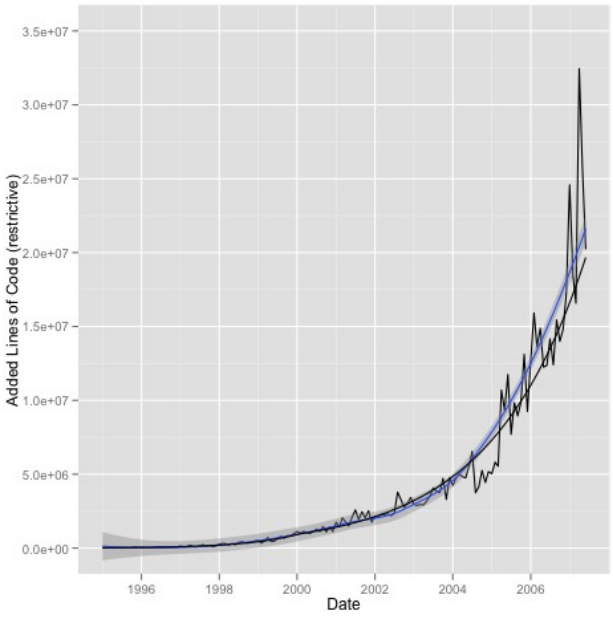

Illustration 61: Exponentialized segmented linear model using GLS against overall added SLoC (restrictive)

On the permissive set, the exponentialized segmented linear model with GLS follows the characteristics of the Loess curve best from a visual standpoint while the restrictive one does not show much difference to the normal segmented linear approach shown above.

6 A model for the growth of active projects per month binned by licenses

The following chapter describes the steps taken towards an analytically closed model of the growth of the number of active projects binned by licenses. The procedure is similar to that in Chapter 5. Chapter 6.1 describes how a Loess-curve was applied to the data to get an insight of the underlying trend. Chapter 6.2 describes how self-starter functions for non-linear models in R where used to find a non-linear model with high Goodness-of-Fit. Chapter 6.3 describes how the most suitable model was analyzed for model violations. Chapter 6.4 the process of log-transforming the respons as a remedy for the model violation discovered in the previous chapter. After various approaches have been tried the chapter concludes that the log-transformation was not an optimal approach and lists ideas for other possible approaches.

6.1 A first glimpse at the number of projects using LOESS for smoothing

A state-factor (0 for dead and 1 for active) was added to each month-window of the cleaned data using the metric for project activity described in 3.1. Then the number of active projects was counted for each month. The results where plotted against a Loess curve to gain a first insight of the underlying trend (Illustration 62 and 63).

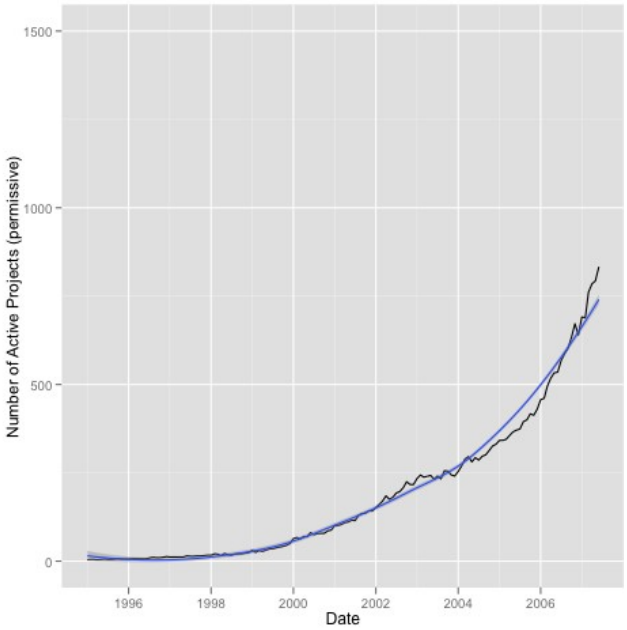

Illustration 62: Number of active projects per month with Loess curve in blue (permissive).

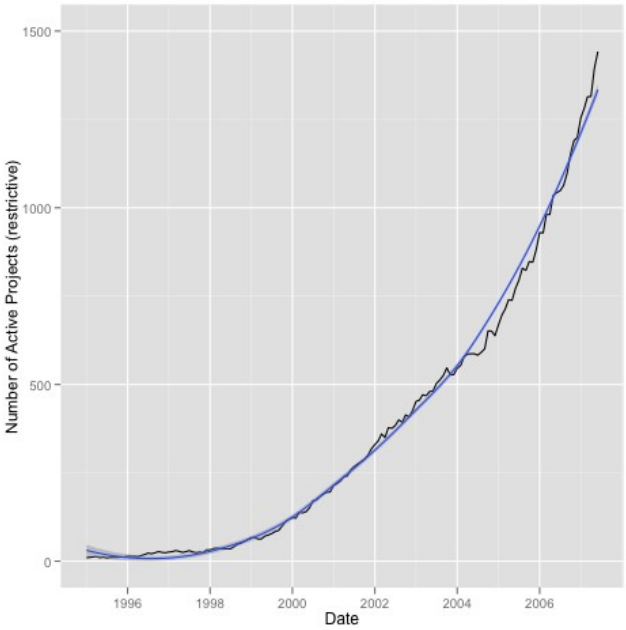

Illustration 63: Number of active projects per month with Loess curve in blue (restrictive).

The plots show a growth-pattern that is similar to the total growth, but with less strong heteroscedasticity. There are also clues that the growth in active projects also consists of two periods for both sets.

6.2 Using self-starter functions models in R to fit nonlinear models

Again the R-function 'nls' was used to fit non-linear models to the data. The results with Pearson's r^2 as a Goodness-of-fit are shown in Table 12:

Model (name of self-starter-function in R)	Goodness-of-fit (Pearson's r^2)	
	Permissive	Restrictive
SSmicmen	-	-
SSbiexp	-	-
SSasymp	-	-
SSasympOff	-	-
SSasympOrig	-	-
SSgompertz	-	0.9933629
SSflp	-	-
SSlogis	-	0.9917365
SSweibull	-	-
Quadratic	0.9709969	0.9895138
Qubic	-	-
SSexp	0.9876267	0.9905861

Table 12: Lists of models tried for growth in active projects with GoF binned by licenses.

The models all show a very good Pearson's *r*, yet the permissive set yielded two additional models, which are plotted below against the data and Loess curve (Illustration 64 and 65):

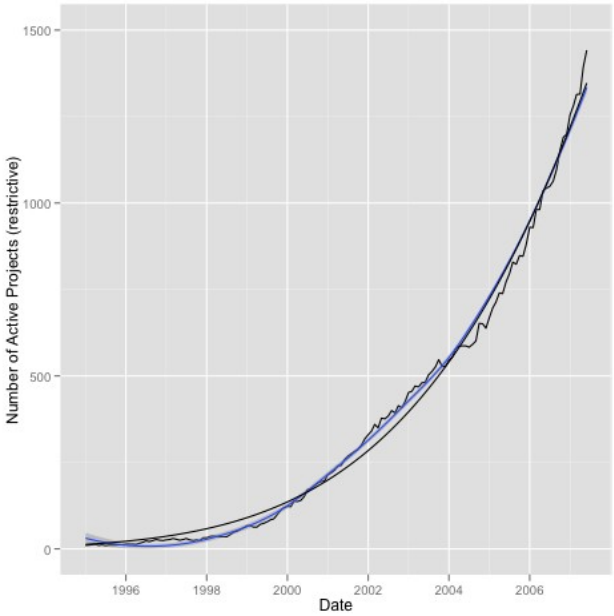

Illustration 64: Gompertz model against number of active projects with Loess curve in blue (restrictive)

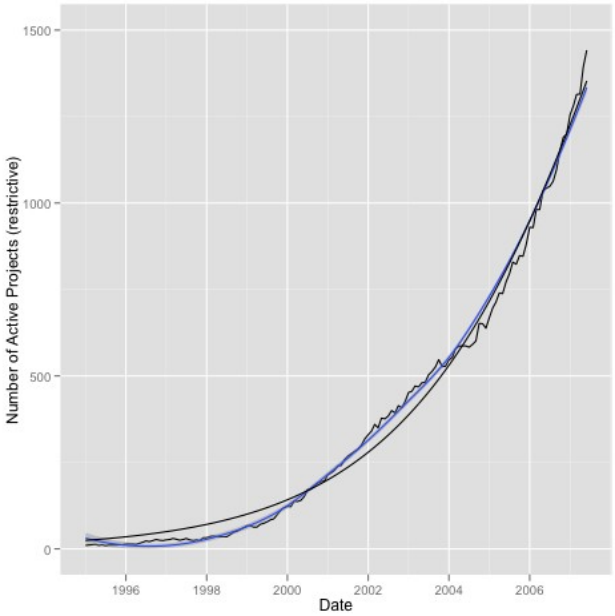

Illustration 65: Logistic model against number of active project with Loess curve in blue (restrictive)

Both models look like they describe the trend well, with the Gompertz model being a little closer in the earlier years. For the sake of parsimony, the models where not used for the comparison.

The following two plots show the quadratic model, which was returned for both sets (Illustration 66 and 67):

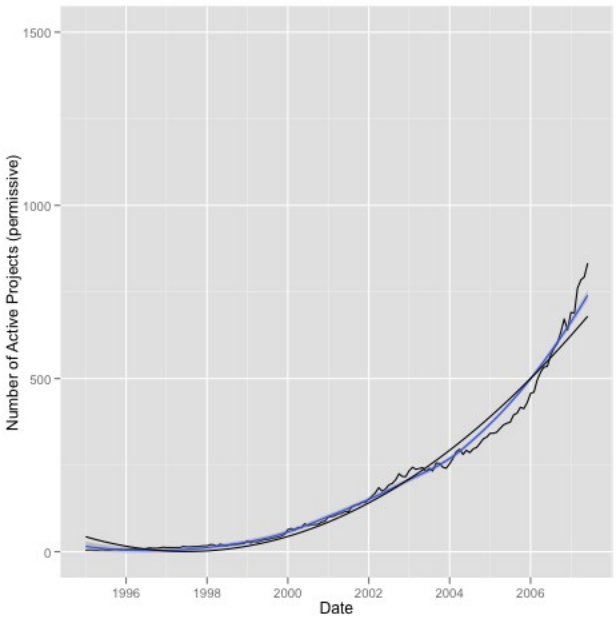

Illustration 66: Quadratic model against number of active projects with Loess curve in blue (permissive)

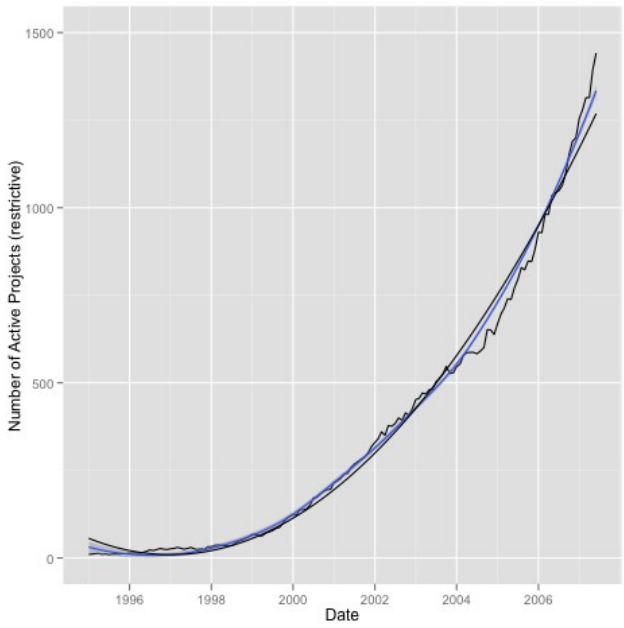

Illustration 67: Quadratic model against number of active projects with Loess curve in blue (restrictive)

Except for the earlier years, the model captures the trend of the data reasonably well in both sets.

The next two plots show the exponential model (Illustration 68 and 69):

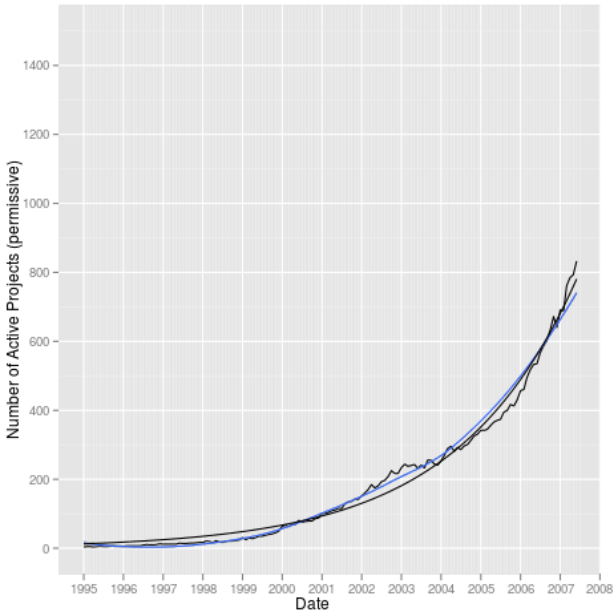

Illustration 68: Exponential model against number of active projects with Loess curve in blue (permissive)

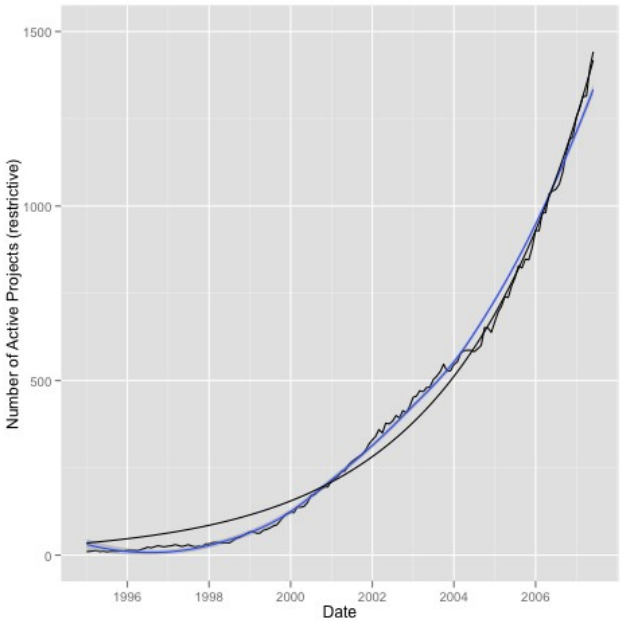

Illustration 69: Exponential model against number of active projects with Loess curve in blue (restrictive)

6.3 A closer look at the exponential model

The exponential model also describes the trend well except for the earlier years in the restrictive set. In the case of heteroscedasticity the exponential model with the segmented log-transformation approach used in 5.4.2 would be a good approach again so the residuals where plotted in Illustration 71 and 71:

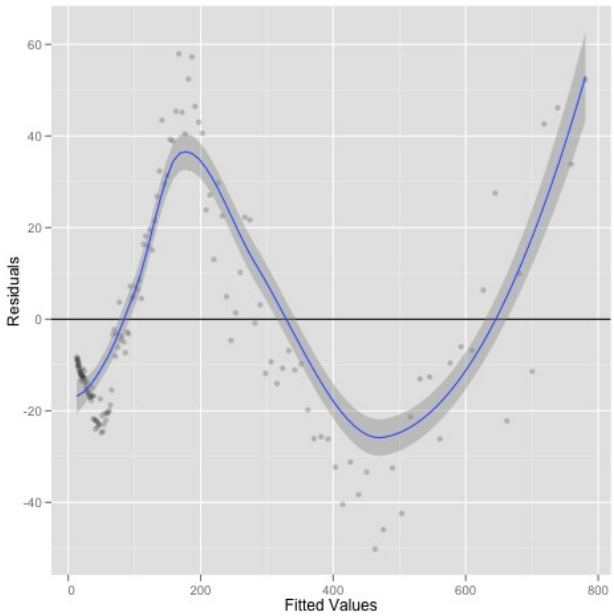

Illustration 71: Fitted values of exponential model against residuals with Loess curve in blue (permissive)

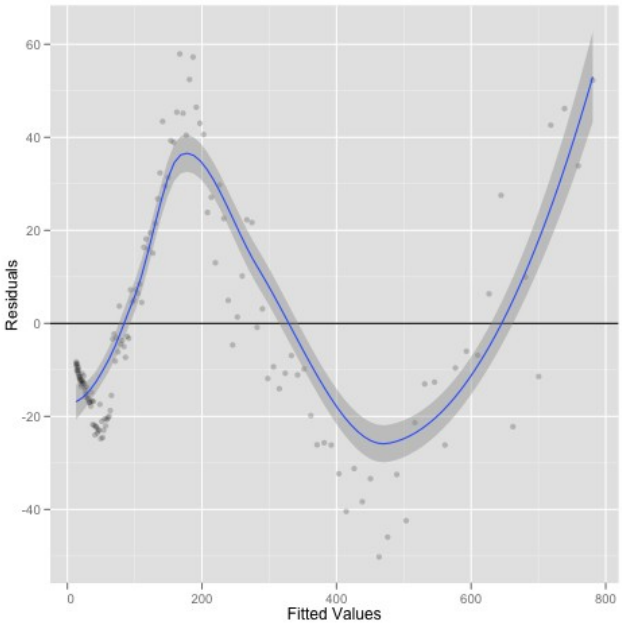

Illustration 70: Fitted values of exponential model against residuals with Loess curve in blue (restrictive)

The residuals showed heteroscedasticity again, but since the model was off in such an extreme way, a plot of the absolute residuals would not have led to any further visual information.

The QQ-plots are shown in Illustration 72 and 73:

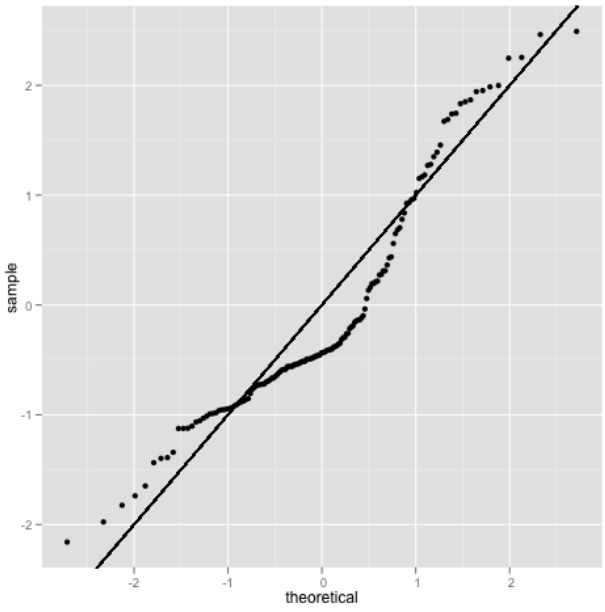

Illustration 72: QQ-plot of exponential model of number of active projects (permissive)

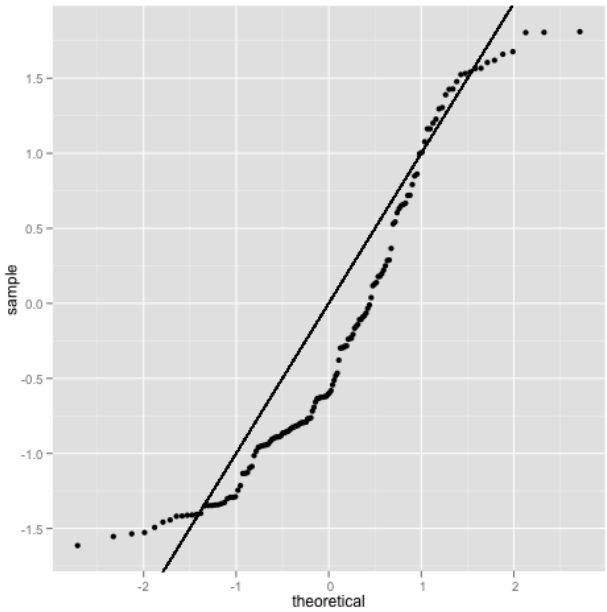

Illustration 73: QQ-plot of exponential model of number of active projects (restrictive)

The residuals strongly deviate from the assumption of normality. Unlike it was the case with the exponential model for the total added SLoC per month the distributions cannot be linearly transformed to a normal distribution.

In the next section, the response will be log-transformed like in the approach for the total added SLoC per month-metric, yet there is some doubt whether it will work out as well as before.

6.4 Log-transformation of the response

After taking the logarithm of the response, the Loess-curve showed that the segmented linear approach could indeed be a good solution once again (Illustration 74 and 75):

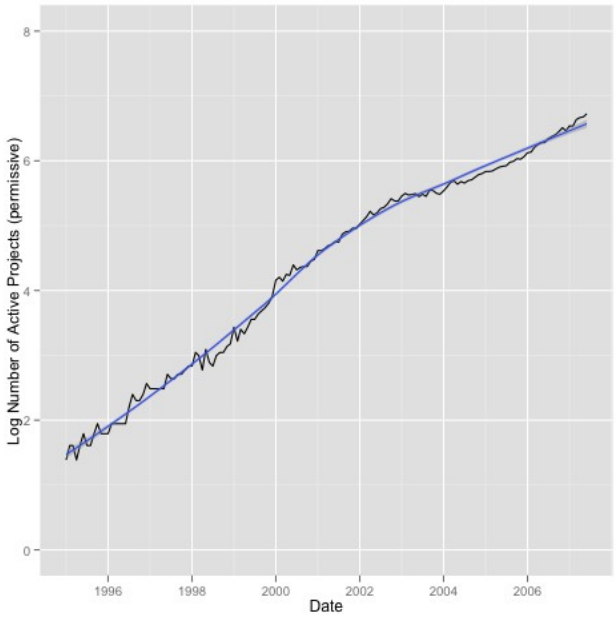

Illustration 74: Number of active projects log-transformed with Loess curve in blue (permissive)

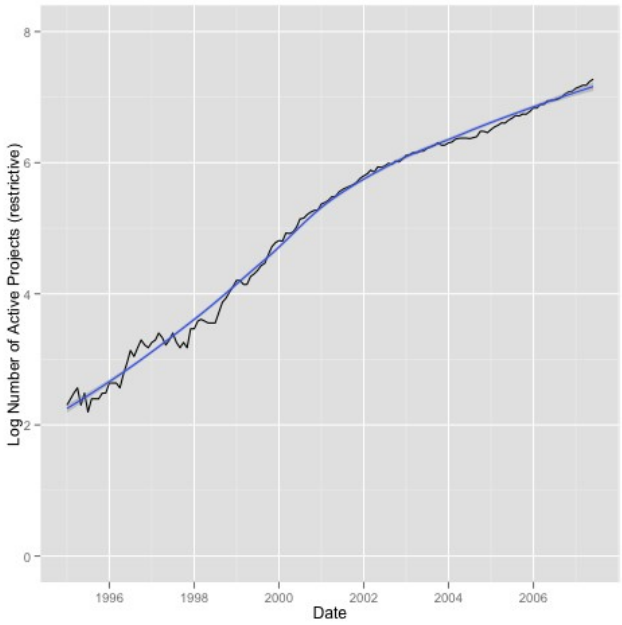

Illustration 75: Number of active projects log-transformed with Loess curve in blue (restrictive)

6.4.1 Linear regression on the log-transformed response

A linear regression was conducted on the log-transformed response that resulted in linear models with an adjusted Pearson's r^2 of 0.976541 for the permissive set and 0.9701497 for the restrictive one. The plots are shown in Illustration 76 and 77:

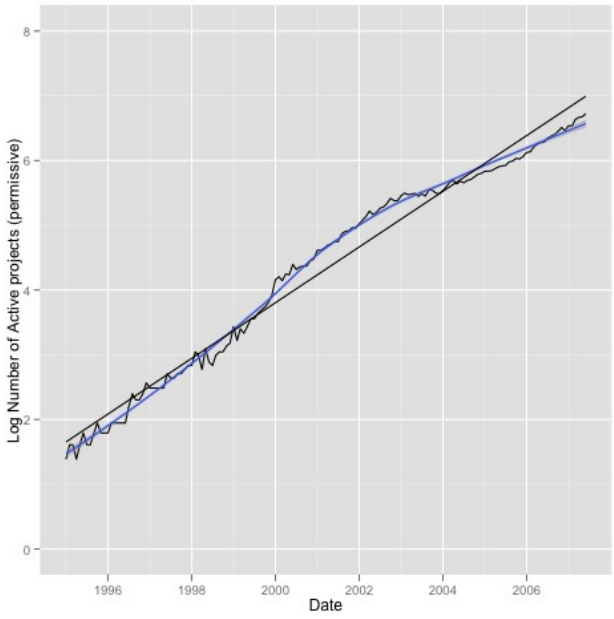

Illustration 76: Linear model against logarithmic number of active projects with Loess curve in blue (permissive)

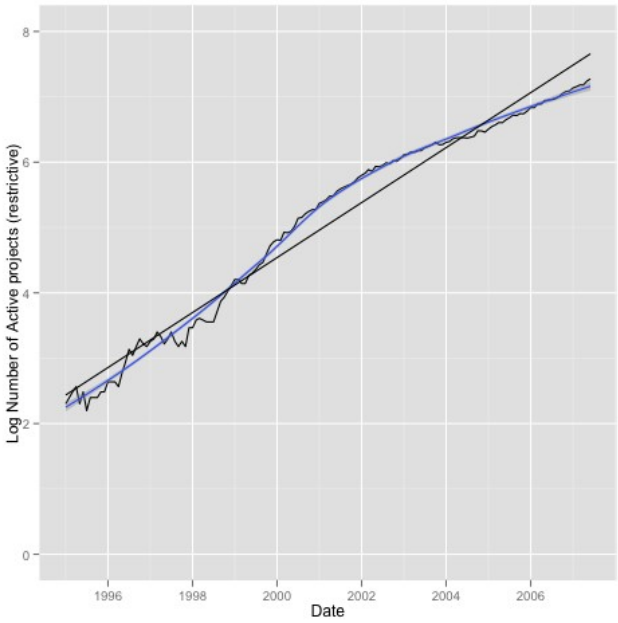

Illustration 77: Linear model against logarithmic number of active projects with Loess curve in blue (restrictive)

The residuals are shown in Illustration 78 and 79 and show that there has been some heteroscedasticity introduced at the beginning (possibly due to the small amount of projects in the database for the earlier years). Also a segmentation is promising again, but there seems to be some additional structure.

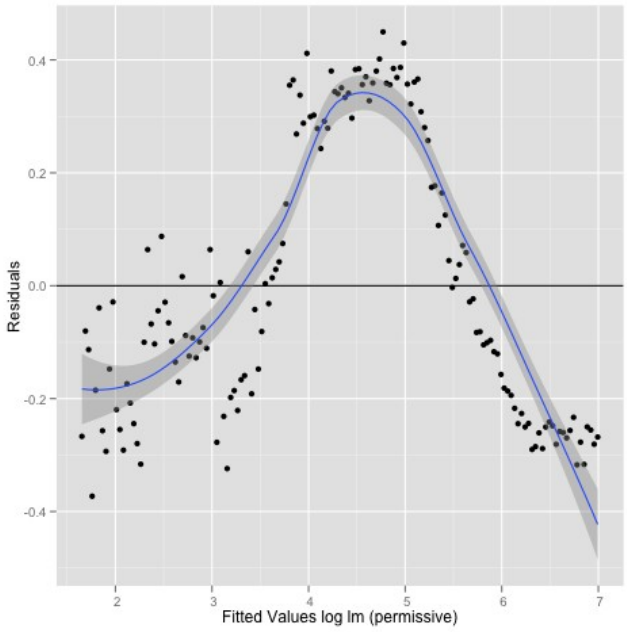

Illustration 78: Fitted values of linear model on logarithmic number of projects against residuals with Loess curve in blue (permissive)

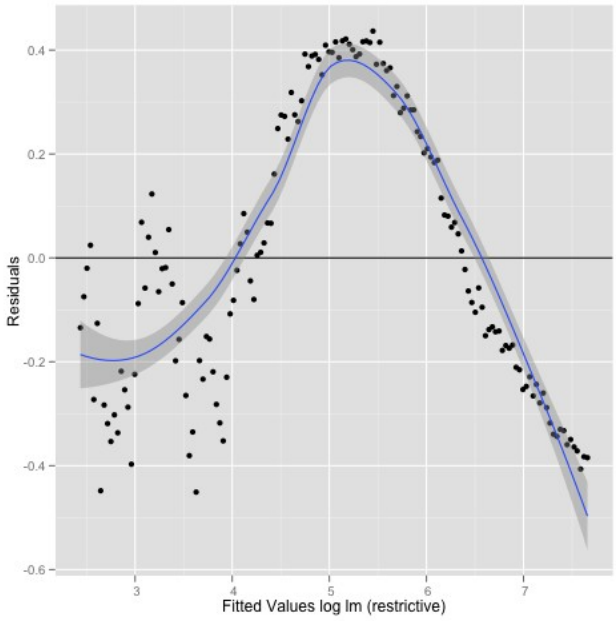

Illustration 79: Fitted values of linear model on logarithmic number of projects against residuals with Loess curve in blue (restrictive)

The QQ-plot is shown in Illustration 80 and 81:

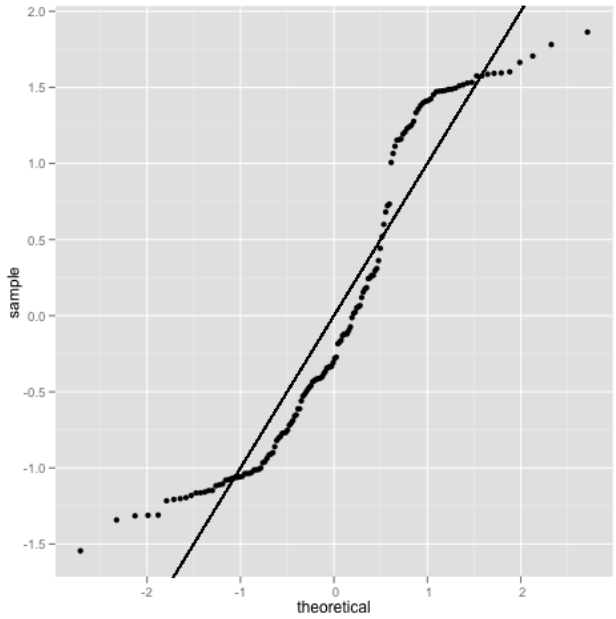

Illustration 80: QQ-plot linear model on log-transformed number of active projects (permissive)

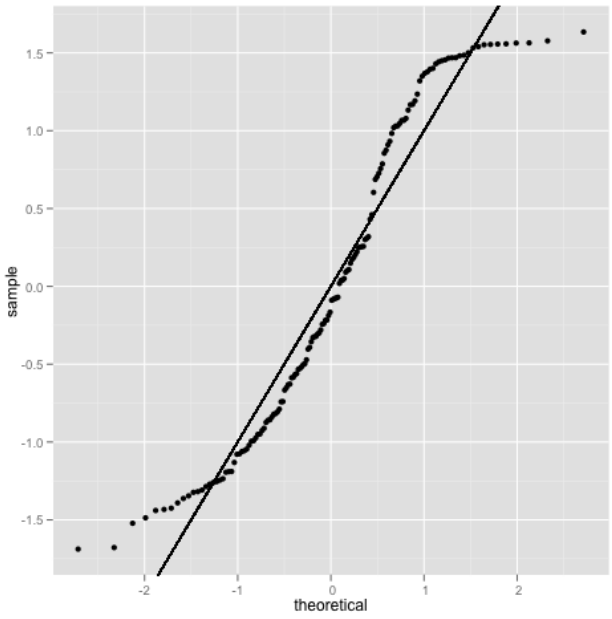

Illustration 81: QQ-plot linear model on log-transformed number of active projects (restrictive)

The distribution of the residuals deviates clearly from a normal distribution.

6.4.2 Segmenting the linear model

The linear models where segmented with one break-point each and resulted in an adjusted Pearson's r^2 of 0.9949866 for the permissive set and 0.993805 for the restrictive. The plots are shown in Illustration 82 and 83:

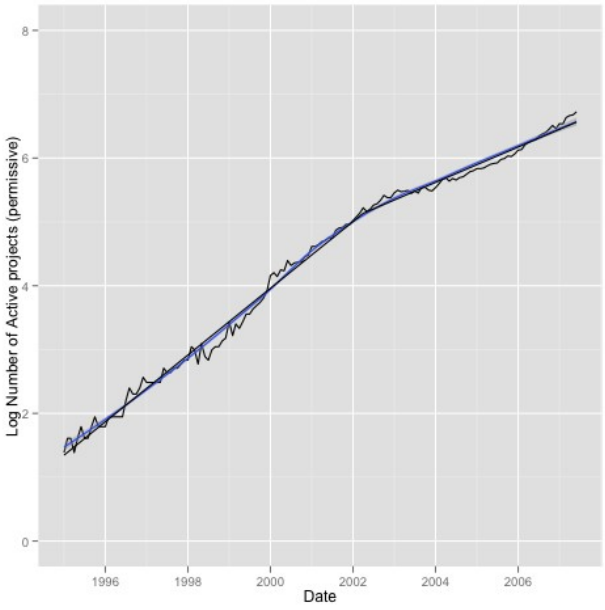

Illustration 82: Segmented linear model against logarithmic number of active projects (permissive)

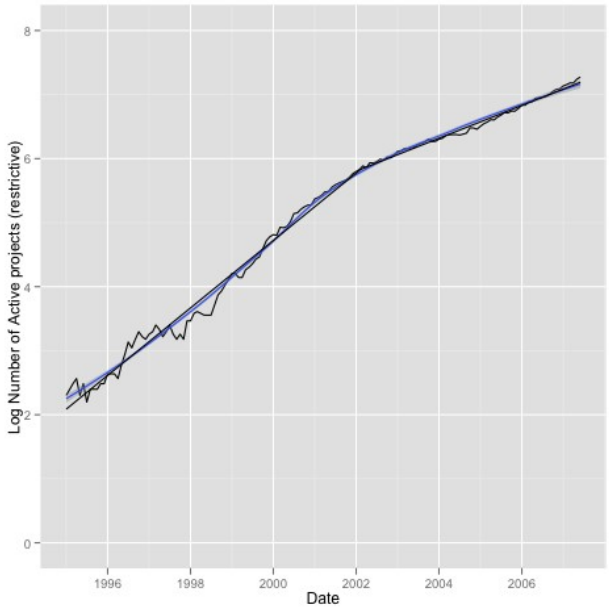

Illustration 83: Segmented linear model against logarithmic number of active projects (restrictive)

The break-points are not as far apart as in the segmented linear models of total added SLoC. The residuals are shown in Illustration 84 and 85:

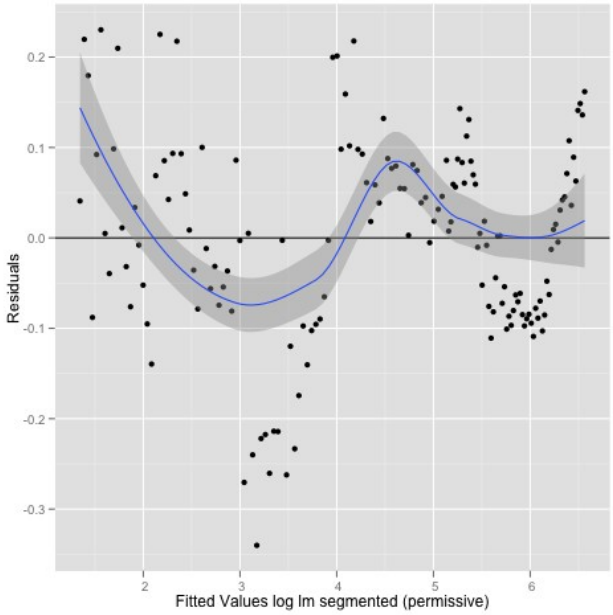

Illustration 84: Fitted values of segmented linear model on logarithmic number of projects against residuals with Loess curve in blue (permissive)

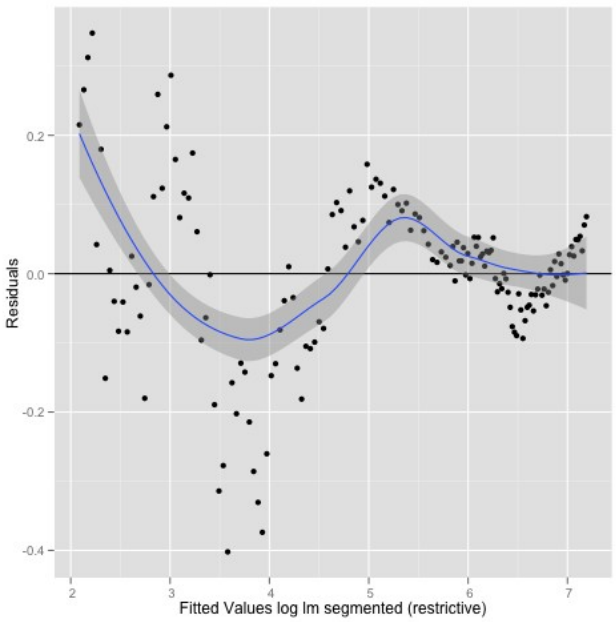

Illustration 85: Fitted values of segmented linear model on logarithmic number of projects against residuals with Loess curve in blue (restrictive)

The residuals show a clear nonlinear structure. The growth appears to be super-linear on the logarithmic reponse.

The QQ-plots are shown in Illustration 86 and 87:

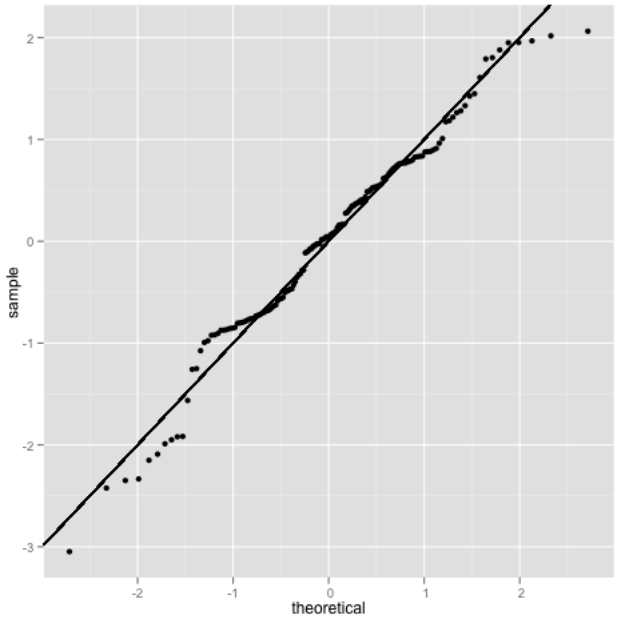

Illustration 86: QQ-plot of segmented linear model on log-transformed number of active projects (permissive)

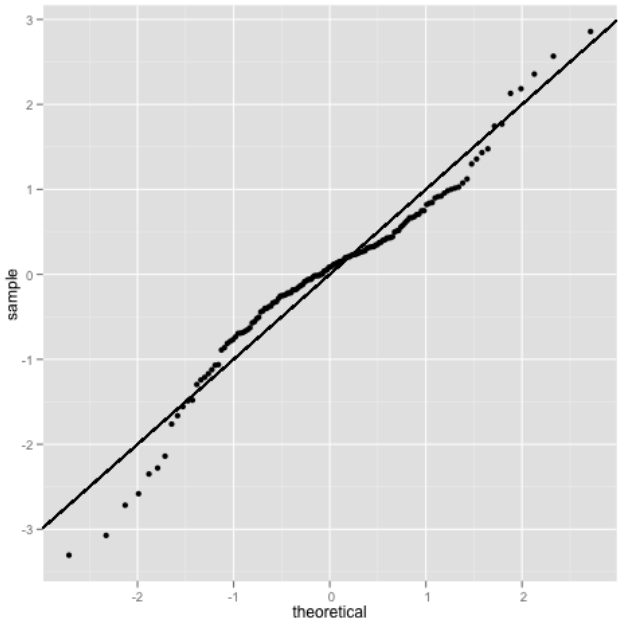

Illustration 87: QQ-plot of segmented linear model on log-transformed number of active projects (restrictive)

The residuals of both the permissive and the restrictive set deviate from the normality assumption, with the restrictive set showing a skewed normal distribution.

No test on correlation was conducted as it can be clearly seen in the residual plots.

6.4.3 Linear regression of quadratic model on the log-transformed response

Fitting a quadratic model resulted in an adjusted Pearson's r^2 of 0.9906921 for the permissive set and 0.988665 for the restrictive. The plots are shown in Illustration 88 and 89:

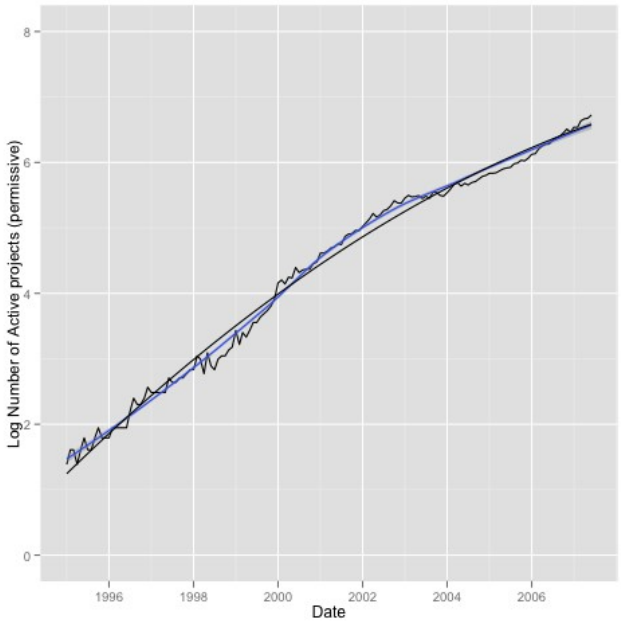

Illustration 88: Quadratic model against logarithmic number of active projects (permissive)

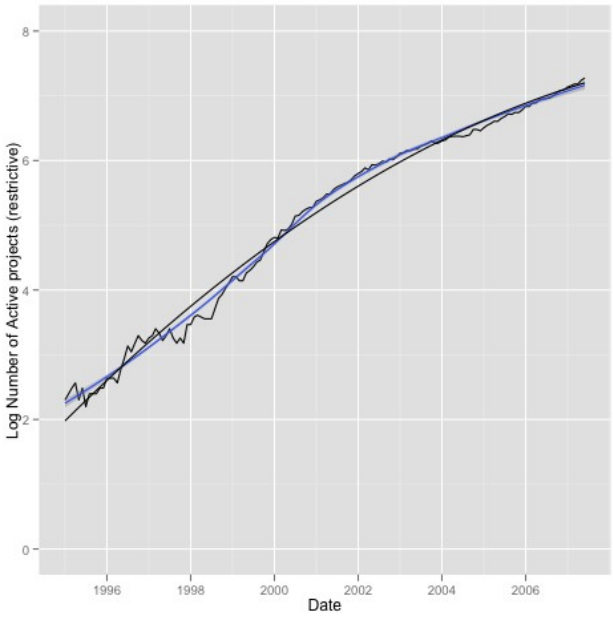

Illustration 89: Quadratic model against logarithmic number of active projects (restrictive)

This model, in contrast, would describe a sub-linear growth on the logarithmic response. The residuals are shown in Illustration 90 and 91:

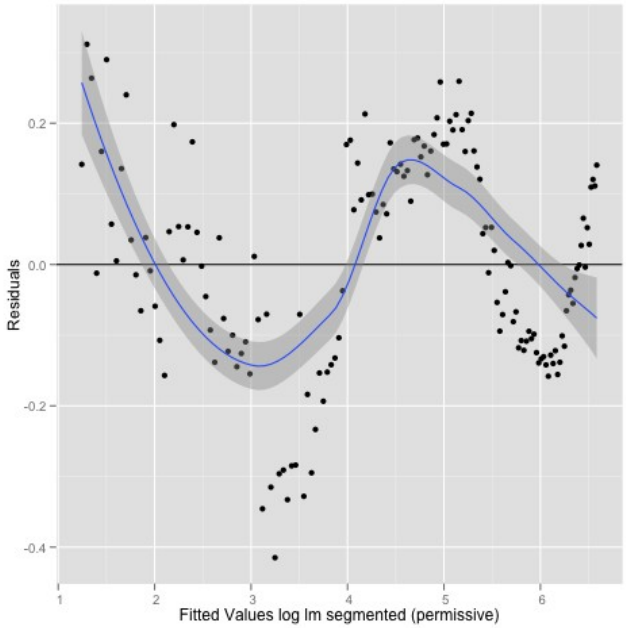

Illustration 90: Fitted values of quadratic model on logarithmic number of projects against residuals with Loess curve in blue (permissive)

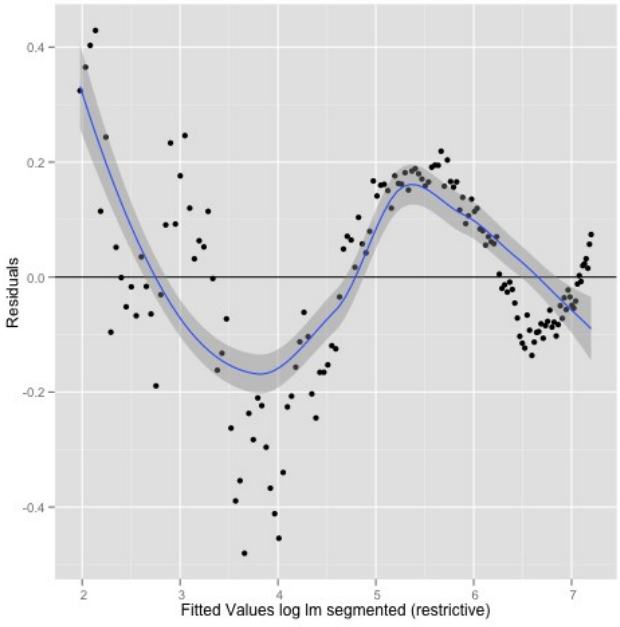

Illustration 91: Fitted values of quadratic model on logarithmic number of projects against residuals with Loess curve in blue (restrictive)

The residuals show a clear structure. The QQ-plots are shown in Illustration 92 and 93:

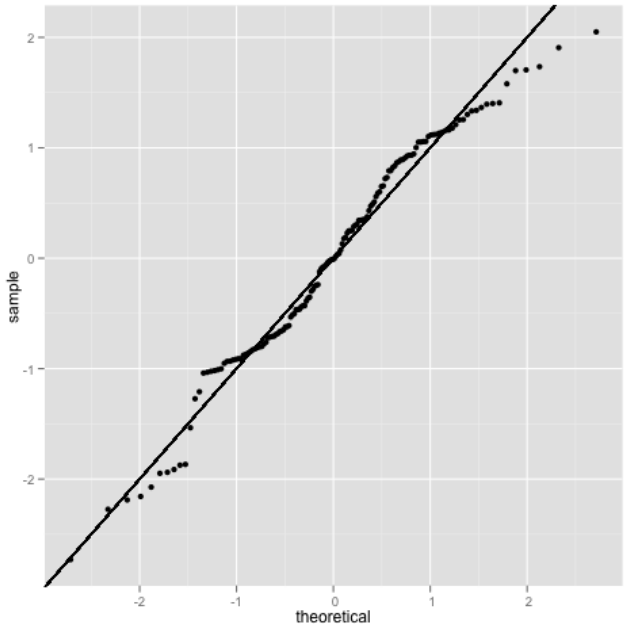

Illustration 92: QQ-plot of quadratic model on log-transformed number of active projects (permissive)

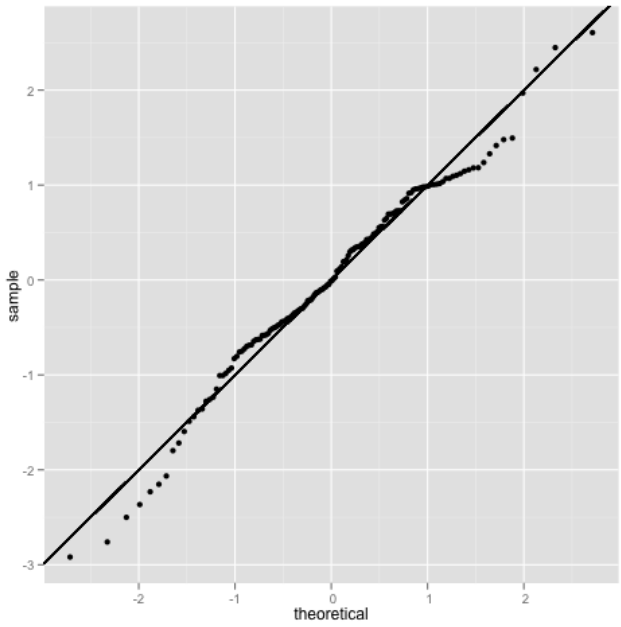

Illustration 93: QQ-plot of quadratic model on log-transformed number of active projects (restrictive)

While the permissive set shows deviation from a normal distribution, the restrictive set is very close. Both the residual plots and the QQ-plots indicate that also the quadratic model requires a segmentation[48].

6.4.4 Segmentation of the quadratic model

The segmentation resulted in models with an adjusted Pearson's r^2 of 0.9965667 for the permissive set and 0.9953521 for the restrictive[49]. The plots are shown in Illustration 94 and 95:

48 The 'segmented' package in R does not support segmentation of linear models from 'ln' with quadratic terms so once again the break-point estimates from the segmented linear regression where used.

49 The highest values of adjusted Pearson's r^2 of the four models on log-transformed response discussed here.

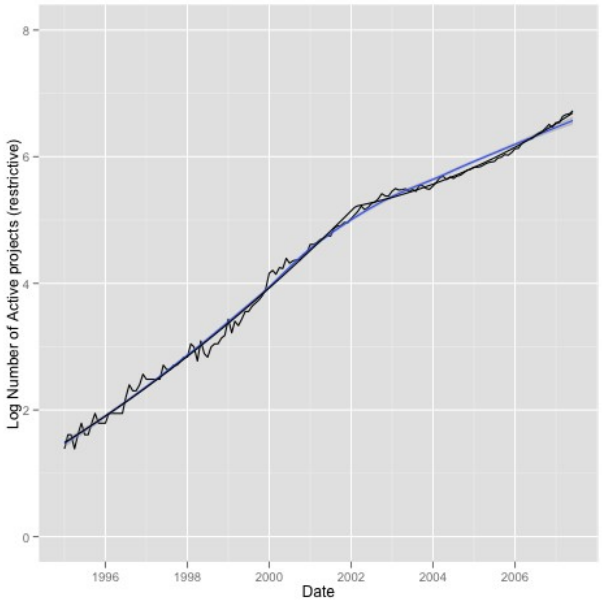

Illustration 94: Segmented quadratic model against logarithmic number of active projects (permissive)

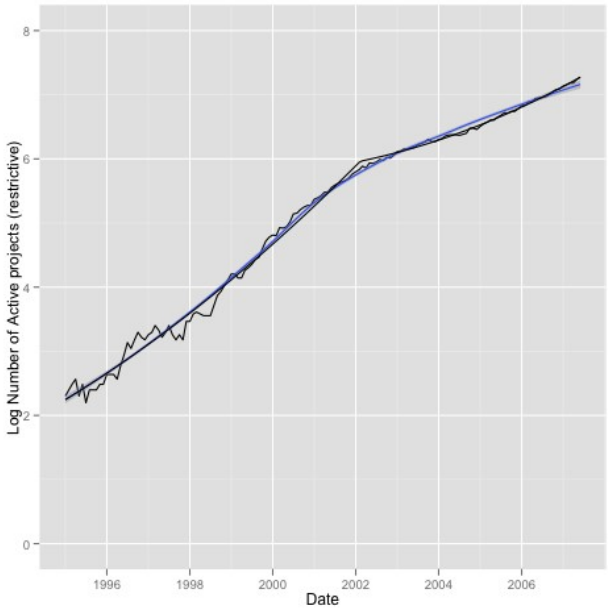

Illustration 95: Segmented quadratic model against logarithmic number of active projects (restrictive)

Visually the quadratic model captures the trend well. For both sets it consists of two segments with super-linear growth.
The residuals are shown in Illustration 96 and 97:

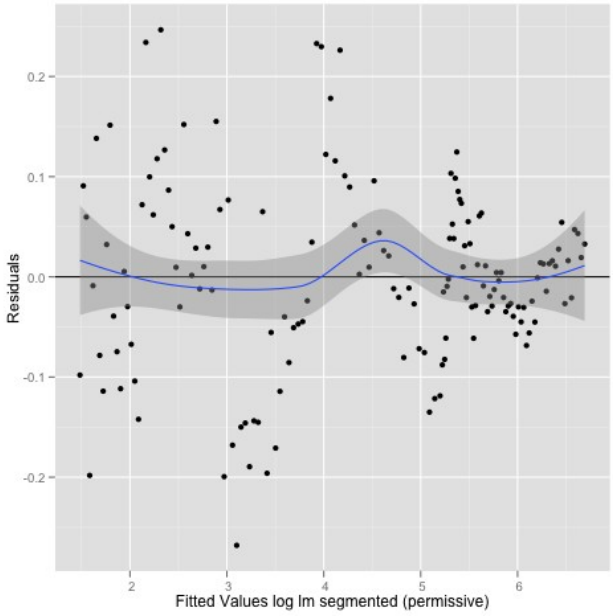

Illustration 96: Fitted values of segmented quadratic model on logarithmic number of projects against residuals with Loess curve in blue (permissive)

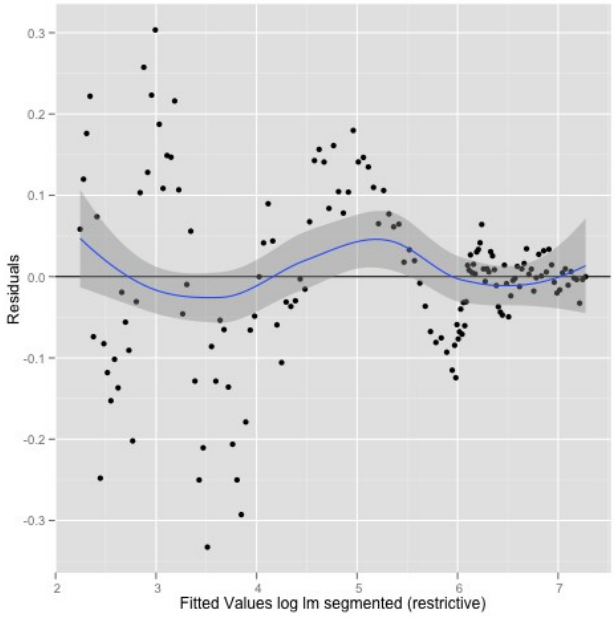

Illustration 97: Fitted values of segmented quadratic model on logarithmic number of projects against residuals with Loess curve in blue (restrictive)

Regarding the structure in the residuals, the quadratic segmented model is an improvement. It also makes the heteroscedasticity a lot clearer.
The QQ-plots are shown in Illustration 98 and 99:

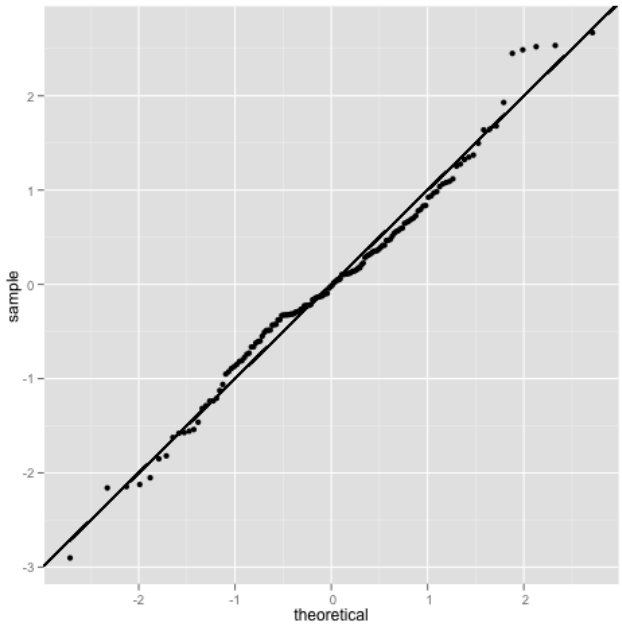

Illustration 98: QQ-plot of segmented quadratic model on log-transformed number of active projects (permissive)

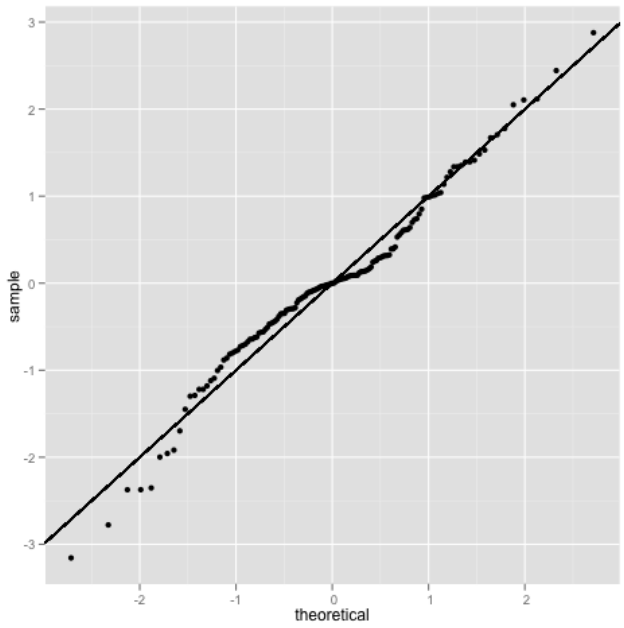

Illustration 99: QQ-plot of segmented quadratic model on log-transformed number of active projects (restrictive)

Both sets still deviate slightly from the normal distribution, but the segmented quadratic approach seems to yield the closest results.

Regarding the heteroscedasticity in the residuals and the deviations of their distributions from the normal distribution the log-transformation seems not to be the best choice for the number of active projects. Using the Box-Cox transformation might be a good ansatz here.

Another possible approach might be to try to fit a segmented sigmoid function directly to the non-transformed data.

7 Normalization of the added SLoC per month by number of active projects

7.1 A first glimpse at the data using Loess for smoothing

The total number of added SLoC was normalized by the number of active projects per month. The results are shown in Illustration 100 and 101[50]:

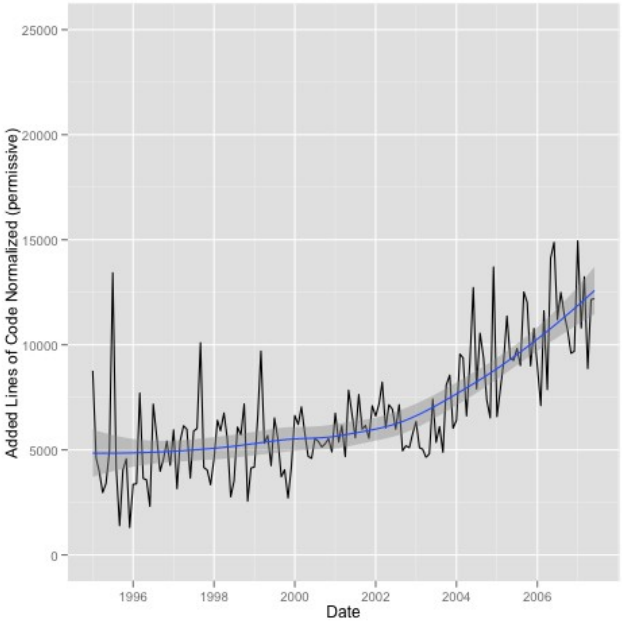

Illustration 100: Added SLoC normalized by number of active projects with Loess curve in blue (permissive).

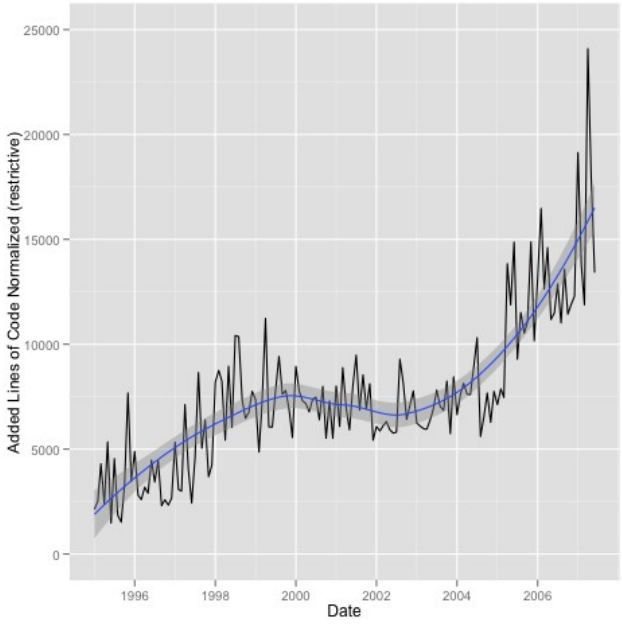

Illustration 101: Added SLoC normalized by number of active projects with Loess curve in blue (restrictive).

In the permissive set, the Loess-curve indicates segmented linear or super-linear growth per project. The restrictive set shows a Loess-curve with changing-point. There could also be a very flat changing-point in the permissive set. The date for the changing-point roughly corresponds with the estimates for the break-points in the segmented models for total added SLoC per month and number of active projects per month.

7.2 Using Self-starter functions in R to fit non-linear models

The semi-automated fitter was used again to fit non-linear models without the requirement of starting-values on the normalized data. The results are listed in Table 13:

50 The plots look a little different from the ones used for outlier detection in 4.2 because for every month, the added SLoC of dead projects are not counted. For the manual outlier detection, the added SLoC of dead projects was important because in the analysis of total growth, the activity of a project was not considered.

Model (name of self-starter-function in R)	Goodness-of-fit (Pearson's r^2)	
	Permissive	Restrictive
SSmicmen	-	-
SSbiexp	-	-
SSasymp	-	-
SSasympOff	-	-
SSasympOrig	-	-
SSgompertz	-	-
SSflp	0.5972122	-
SSlogis	-	-
SSweibull	-	-
Quadratic	0.5869503	0.5906754
Qubic	-	-
SSexp	0.5429717	0.5984381

Table 13: Lists of models tried for growth per active project with GoF binned by licenses.

No function fit the data good enough for consideration, probably due to the amount of noise present. For the restrictive set, a number of non-linear functions from the double-logistic family where tried with no success. For the permissive set, a segmented linear model would yield an adjusted Pearson's r^2 of 0.5958.

No analytically closed model could be found for the growth-per-project approach. Yet the plot of the data with Loess curves gives hints on what might have happened around the time of the break-points of the total growth and the number of active projects. The Loess curves also suggest that the growth-per-project is either super-linear for both sets or segmented linear for the permissive and super-linear for the restrictive set.

8 Discussion of results and impacting factors

In the following chapter the results of the analysis and possible impacting factors are discussed.

The results from Chapter 5 confirm the results from Deshpande and Riehle (2008) [24] that open source in total is growing at an exponential rate for both the restrictive and the permissive set in the time from 1995 to the middle of 2007. Yet the growth-pattern for both sets can be divided in two periods of growth. In the first period, ranging from 1995 to roughly 2000/2002 the restrictive set shows a significantly faster growth than the permissive. In the second period, which is ranging up until the middle 2007, the growth of both sets has slowed down. This effect is a lot stronger for the restrictive set resulting in the indication that for the second period, the permissive projects in total are growing faster.
For the number of active projects per month, an analytically closed model for the growth could not be found, but break-points where estimated around 2002. For the average growth-per-project, no analytically closed model could be found, either. But the Loess-curves indicate a change for both permissive and restrictive around 2001/2001. So, what happened around that date? Let us re-visit the time bar from Chapter 2.2 with some additional dates added (Illustration 102)[51]:

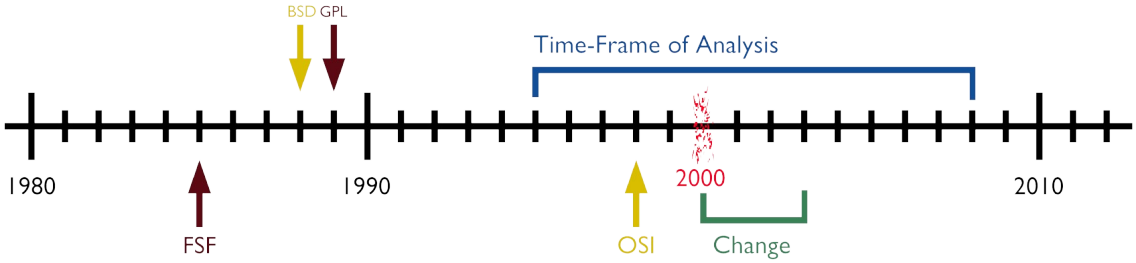

Illustration 102: Time bar with date of foundation of FSF and OSI, date of publication of the first permissive (BSD) and restrictive (GPL) licenses, burst of the dotcom bubble in red, the time-frame of the analysis and the time-frame of change in the growth of the two bins.

The year 2000 was the year of the so-called 'dotcom bubble' where the overheated hi-tech stock market crashed in the US and worldwide, forcing many software firms into bankruptcy. It took three years for the stock market to stabilize again[52]. A time-frame that coincides with the period of negative growth that can be seen in the Loess-curve of the average growth-per-project of the restrictive set (Illustration 101). Also, the break-points for the segmentations of the total growth in SLoC and number of active projects for both the permissive and restrictive set are located in that time-frame.
Considering the burst of the 'dotcom bubble' as the cause of the decline in total growth for both bins, the questions arises why it had less effect on the permissive set[53]. Until 1998, the movement that is today know as the 'open source movement' was largely driven by the FSF and it's social and ethical agenda. What today is mostly referred to as 'open source software' was known as 'free software' during that time. With the formation of the OSI, the main perception changed drastically. 'Open source' in contrast to 'free software' is a practical business-oriented approach that got huge attention in both media and industry. The OSI does not solely advocate permissive licenses but

51 Note that the choice of the same color for the introduction of the BSD license and the date of foundation of the OSI are not meant to denote that the OSI is specifically pushing the BSD license (which it does not), but rather that the OSI is an organization that is advising the use of both permissive and restrictive licenses while the FSF favor the use of restrictive licenses in most cases.
52 See http://news.bbc.co.uk/2/hi/business/8558257.stm for a graph of the Nasdaq index from 1995 to 2010.
53 In case of growth-per-project no effect at all, see Illustration 100.

rather allows a neutral discussion on the pros and cons of licensing from a practical point of view without an ideological bias, something that was lacking before it's foundation.
Another possible impact factor why the permissive projects did not suffer that much during the time from 2000 to 2003 might be the people and communities behind permissive and restrictive projects. The permissive-licensed projects might have simply handled the change in the IT-ecosystem better than the restrictive ones. To back this up, the number of commiters would have been required to be taken into account.
It could also be the case that open source practitioners learned from mistakes made before and during the burst and partly shifted towards permissive licensing.

9 Limitations of analysis

The quantitative analyses has shortcomings in regard to the database used. Deshpande and Riehle (2008) [24] have been using an earlier snapshot of the same database for their study and list shortcomings and limitations that also apply to this analysis (quotes in 10pt font size):

- *Sample size:* After the cleanup process, the sample constituted of 1861 projects in the category 'permissive' and 3257 projects in the category 'restrictive'. The real number of active projects in both categories was much larger during the analyzed time-frame. Yet the estimated 30% of active open source projects hold by the database can still be considered "relevant for analyzing trends and patterns in open source growth".
- *Data incompleteness*: "Some amount of revision control information in open source projects has already been lost forever, as projects have moved on from no configuration management (CM) to CM with CVS and on to other CM tools, frequently dropping the history with each move. Thus, the project history for each project is not always complete. However, for a current project, we have the most recent history, which is what is most relevant for our analysis. Thus, the lack of some of the early histories of some of the open source projects has little effect on the validity of our conclusions."
  This effect can still be an issue here since the database has a selection-bias in case of older projects. Projects that did not succeed are simply not included up to a certain point (the collection of data started in 2005). However this does not effect the results regarding the differences in growth between the permissive and the restrictive set since the selection-bias does not differentiate between licenses.
- *Project source:* "A current limitation of Ohloh is that it only connects to CVS, Subversion and Git source code repositories. We believe that this limitation is not a big issue for our purposes because almost all open source projects are maintained in one of these repositories and our sample size can be considered representative."
- *Copy and paste:* The database does not account for copy and paste. The cleanup process employed in this thesis only accounts for copy and paste in the manual part, where only the most extreme cases where filtered out. While this is not an issue when looking at the overall trend, copy and paste introduces a bias towards restrictive-licensed projects because a restrictive project can incorporate code from a permissive one but not vice-versa. To analyze the influence of this bias is a suggestion for further research.

10 Conclusion and further research

10.1 Conclusion

This thesis analyzed the growth of a large sample of open source projects in a time-frame from 1995 to the middle of 2007 binned by licenses. On the metric of total growth in SLoC, in the period from 1995 to roughly 2000/2001 the restrictive set showed a significantly higher growth. In the period from 2001 to the midst of 2007, there is an indication that the permissive projects grew faster in total, yet the growth for both types of projects was slower in the second period.
On the metric of number of active projects, break-points where estimated for around 2002, yet no analytically closed model could be found to describe the growth-pattern.
On the metric of growth-per-project, no analytically closed model was found either, but smoothing the data with a Loess-curve indicates that the growth-per-project of the restrictive set was declining during a period from 2000 to 2003 but not the growth-per-project of the permissive set.
As a possible impact factor for this change, the burst of the 'dotcom bubble', the foundation of the OSI and other factors where discussed.

10.2 Further Research

The research in this thesis is based on a database that is not publicly available. Further research is required to confirm the results for example on the FLOSSmole database.

Further research is also required to find out whether the trend is continuing until today or whether things have changed recently. The Ohloh database snapshot used in this theses only has robust data up until the midst of 2007.

The research only yielded analytically closed models for the total growth binned by licenses. Further research is required to find models for the growth in number of active projects and growth-per-project. Further metrics to consider are number of commiters and number of commiters per project.

In this thesis, growth was binned by permissive and restrictive licenses. But the discovered periods of growth make this work also valuable for research of the total growth of open source. Further research could try to confirm the periods for a sample of projects that are not binned by licenses and that includes projects that are licensed semi-restrictive like LGPL which have not been considered in this thesis for reasons of simplicity.

10.3 Acknowledgements

I would like to thank Prof. Dr. Dirk Riehle for the thesis opportunity, Carsten Kolassa for his ongoing support and Wolfgang Maurer for valuable input.

Abbreviations

AIC	Aikake Information Criterion
BIC	Bayesian Information Criterion
FSF	Free Software Foundation
GoF	Goodness-of-Fit
GLS	Generalized Least-Squares
GPL	GNU General Public License
GNU	Gnu's not Unix
LGPL	GNU Lesser General Public License
OSI	Open Source Initiative
OSS	Open Source Software
SLoC	Source Lines of Code

Illustration Index

Illustration 1: Time bar of the analyzed time period and the introduction of the prototypes of the two license-types used for binning. 3
Illustration 2: Raw SLoC added permissive. 8
Illustration 3: Raw SLoC added restrictive. 8
Illustration 4: Added SLoC without multiple repositories permissive. 9
Illustration 5: Added SLoC without multiple repositories restrictive. 9
Illustration 6: Added SLoC without multiple repositories and initial commit (permissive). 10
Illustration 7: Added SLoC without multiple repositories and initial commit (restrictive). 10
Illustration 8: Normalized data for outlier detection (permissive) 10
Illustration 9: Normalized data for outlier detection (restrictive) 10
Illustration 10: Cleaned data (permissive) 11
Illustration 11: Cleaned data (restrictive) 11
Illustration 12: Average added lines of code per project (permissive) 12
Illustration 13: Average added lines of code per project (restrictive) 12
Illustration 14: Cleaned data with Loess curve in blue (permissive) 13
Illustration 15: Cleaned data with Loess curve in blue (restrictive) 13
Illustration 16: Quadratic model against overall added SLoC (permissive) 14
Illustration 17: Quadratic model against overall added SLoC (restrictive) 14
Illustration 18: Cubic model against overall added SLoC (permissive) 15
Illustration 19: Cubic model against overall added SLoC (permissive) 15
Illustration 20: Exponential model against overall added SLoC (permissive) 15
Illustration 21: Exponential model against overall added SLoC (restrictive) 15
Illustration 22: Fitted values of exponential model against residuals with Loess curve in blue (permissive) 16
Illustration 23: Fitted values of exponential model against residuals with Loess curve in blue (restrictive) 16
Illustration 24: Fitted values of exponential model against absolute residuals with Loess curve in blue (permissive) 17
Illustration 25: Fitted values of exponential model against residuals with Loess curve in blue (restrictive) 17
Illustration 26: QQ-plot exponential model (permissive) 18
Illustration 27: QQ-plot exponential model (restrictive) 18

Illustration 28: Logarithmic added SLoC with Loess curve in blue (permissive)..... 19
Illustration 29: Logarithmic added SLoC with Loess curve in blue (restrictive)..... 19
Illustration 30: Linear model against logarithmic added SLoC (permissive)..... 20
Illustration 31: Linear model against logarithmic added SLoC (restrictive)..... 20
Illustration 32: Fitted values of linear model on logarithmic data against residuals with Loess curve in blue (permissive)..... 20
Illustration 33: Fitted values of linear model on logarithmic data against residuals with Loess curve in blue (restrictive)..... 20
Illustration 34: QQ-plot linear model on log-transformed response (permissive)..... 21
Illustration 35: QQ-plot linear model on log-transformed response (restrictive)..... 21
Illustration 36: Segmented linear model against logarithmic added SLoC (permissive)..... 21
Illustration 37: Segmented linear model against logarithmic added SLoC (restrictive)..... 21
Illustration 38: Fitted values of segmented linear model on logarithmic data against residuals with Loess curve in blue (permissive)..... 22
Illustration 39: Fitted values of segmented linear model on logarithmic data against residuals with Loess curve in blue (restrictive)..... 22
Illustration 40: QQ-plot segmented linear model on log-transformed response (permissive)..... 23
Illustration 41: QQ-plot segmented linear model on log-transformed response (restrictive)..... 23
Illustration 42: Correlogram of the studentized residuals of the segmented linear model (restrictive) 24
Illustration 43: Correlogram of the studentized residuals of the segmented linear model (permissive) 24
Illustration 44: Segmented linear model against logarithmic added SLoC using GLS (permissive) 26
Illustration 45: Segmented linear model against logarithmic added SLoC using GLS (restrictive) 26
Illustration 46: Fitted values of segmented linear model using GLS on logarithmic data against residuals with Loess-curve in blue (permissive)..... 26
Illustration 47: Fitted values of segmented linear model using GLS on logarithmic data against residuals with Loess-curve in blue (restrictive)..... 26
Illustration 48: QQ-plot segmented linear GLS model on log-transformed response (permissive)... 27
Illustration 49: QQ-plot segmented linear GLS model on log-transformed response (restrictive).... 27
Illustration 50: Slope estimates of the linear models on log-transformed response with 95% confidence intervals..... 28
Illustration 51: Slope estimates of the first period of the segmented linear models on log-transformed response with 95% confidence intervals..... 29

Illustration 52: Slope estimates of the second period of the segmented linear models on log-transformed response with 95% confidence intervals. 29
Illustration 53: Estimates of the break-points of the segmented models alongside the models and the data on log transformed response. Red indicates the restrictive set and blue the permissive. 30
Illustration 54: Slope estimates of the first period of the segmented linear models fit by GLS on log-transformed response with 95% confidence intervals. 31
Illustration 55: Slope estimates of the second period of the segmented linear models fit by GLS on log-transformed response with 95% confidence intervals. 31
Illustration 56: Exponentialized linear model against overall added SLoC (permissive) 34
Illustration 57: Exponentialized linear model against overall added SLoC (restrictive) 34
Illustration 58: Exponentialized segmented linear model against overall added SLoC (permissive) 34
Illustration 59: Exponentialized segmented linear model against overall added SLoC (restrictive) 34
Illustration 60: Exponentialized segmented linear model using GLS against overall added SLoC (permissive) 35
Illustration 61: Exponentialized segmented linear model using GLS against overall added SLoC (restrictive) 35
Illustration 62: Number of active projects per month with Loess curve in blue (permissive). 36
Illustration 63: Number of active projects per month with Loess curve in blue (restrictive). 36
Illustration 64: Gompertz model against number of active projects with Loess curve in blue (restrictive) 37
Illustration 65: Logistic model against number of active project with Loess curve in blue (restrictive) 37
Illustration 66: Quadratic model against number of active projects with Loess curve in blue (permissive) 38
Illustration 67: Quadratic model against number of active projects with Loess curve in blue (restrictive) 38
Illustration 68: Exponential model against number of active projects with Loess curve in blue (permissive) 39
Illustration 69: Exponential model against number of active projects with Loess curve in blue (restrictive) 39
Illustration 70: Fitted values of exponential model against residuals with Loess curve in blue (restrictive) 40
Illustration 71: Fitted values of exponential model against residuals with Loess curve in blue (permissive) 40

Illustration 72: QQ-plot of exponential model of number of active projects (permissive).....40
Illustration 73: QQ-plot of exponential model of number of active projects (restrictive).....40
Illustration 74: Number of active projects log-transformed with Loess curve in blue (permissive)..41
Illustration 75: Number of active projects log-transformed with Loess curve in blue (restrictive)…41
Illustration 76: Linear model against logarithmic number of active projects with Loess curve in blue (permissive)..... 42
Illustration 77: Linear model against logarithmic number of active projects with Loess curve in blue (restrictive)..... 42
Illustration 78: Fitted values of linear model on logarithmic number of projects against residuals with Loess curve in blue (permissive).....42
Illustration 79: Fitted values of linear model on logarithmic number of projects against residuals with Loess curve in blue (restrictive)..... 42
Illustration 80: QQ-plot linear model on log-transformed number of active projects (permissive). .43
Illustration 81: QQ-plot linear model on log-transformed number of active projects (restrictive)....43
Illustration 82: Segmented linear model against logarithmic number of active projects (permissive) 44
Illustration 83: Segmented linear model against logarithmic number of active projects (restrictive) 44
Illustration 84: Fitted values of segmented linear model on logarithmic number of projects against residuals with Loess curve in blue (permissive).....44
Illustration 85: Fitted values of segmented linear model on logarithmic number of projects against residuals with Loess curve in blue (restrictive)..... 44
Illustration 86: QQ-plot of segmented linear model on log-transformed number of active projects (permissive)..... 45
Illustration 87: QQ-plot of segmented linear model on log-transformed number of active projects (restrictive)..... 45
Illustration 88: Quadratic model against logarithmic number of active projects (permissive).....46
Illustration 89: Quadratic model against logarithmic number of active projects (restrictive).....46
Illustration 90: Fitted values of quadratic model on logarithmic number of projects against residuals with Loess curve in blue (permissive).....46
Illustration 91: Fitted values of quadratic model on logarithmic number of projects against residuals with Loess curve in blue (restrictive)..... 46
Illustration 92: QQ-plot of quadratic model on log-transformed number of active projects (permissive)..... 47

Illustration 93: QQ-plot of quadratic model on log-transformed number of active projects (restrictive)..... 47
Illustration 94: Segmented quadratic model against logarithmic number of active projects (permissive)..... 48
Illustration 95: Segmented quadratic model against logarithmic number of active projects (restrictive)..... 48
Illustration 96: Fitted values of segmented quadratic model on logarithmic number of projects against residuals with Loess curve in blue (permissive)..... 48
Illustration 97: Fitted values of segmented quadratic model on logarithmic number of projects against residuals with Loess curve in blue (restrictive).....48
Illustration 98: QQ-plot of segmented quadratic model on log-transformed number of active projects (permissive)..... 49
Illustration 99: QQ-plot of segmented quadratic model on log-transformed number of active projects (restrictive).....49
Illustration 100: Added SLoC normalized by number of active projects with Loess curve in blue (permissive)..... 50
Illustration 101: Added SLoC normalized by number of active projects with Loess curve in blue (restrictive)..... 50
Illustration 102: Time bar with date of foundation of FSF and OSI, date of publication of the first permissive (BSD) and restrictive (GPL) licenses, burst of the dotcom bubble in red, the time-frame of the analysis and the time-frame of change in the growth of the two bins.....52

Index of Tables

Table 1: Licenses by Type. Multiple versions of a license are counted as one. For example GPL v1, v2 and v3 are listed as GPL only. 4
Table 2: List of models tried for total growth with Goodness-of-Fit binned by license. 14
Table 3: Autocorrelation and Durbin-Watson-Statistic for the segmented linear models up to lag 3 24
Table 4: Comparison of model-selection criteria for the generalized least-squares fit with and without provision of correlation (segmented approach). 25
Table 5: Comparison of non-segmented linear models on log-transformed response for the restrictive and permissive set and confidence intervals for the slope. 28
Table 6: Comparison of the slope of the segmented linear models on log-transformed response for the restrictive and permissive set including confidence intervals. 29
Table 7: Estimated break-points for the segmented linear model on log-transformed response and 95% confidence intervals. 30
Table 8: Comparison of the slope of the segmented linear models using GLS on log-transformed response for the restrictive and permissive set including confidence intervals. 31
Table 9: The differences in growth binned by model and license-type with confidence intervals. . . 32
Table 10: Comparison of the linear models transformed to the non-logarithmic scale. 33
Table 11: Error-bias of the transformed linear models 33
Table 12: Lists of models tried for growth in active projects with GoF binned by licenses. 37
Table 13: Lists of models tried for growth per active project with GoF binned by licenses. 51

Index of Literature

[1] A. Aksulu and M. Wade, "A Comprehensive Review and Synthesis of Open Source Research," *October*, vol. 11, no. 11, pp. 576-656, 2010.
URL: http://aisel.aisnet.org/jais/vol11/iss11/6/

[2] J. Lerner and J. Tirole, "Some simple economics of open source," *The journal of industrial economics*, vol. L, no. 2, 2002.
URL: http://onlinelibrary.wiley.com/doi/10.1111/1467-6451.00174/abstract

[3] GNU Project.
URL: http://www.gnu.org/
(last visit 2012.02.07)

[4] Free Software Foundation.
URL: http://www.fsf.org/
(last visit 2012.02.07)

[5] GNU Manifesto.
URL: http://www.gnu.org/gnu/manifesto.en.html
(last visit 2012.02.07)

[6] "What is the Free Software Foundation?," *GNU'S Bulletin*, vol. 1, no.1, p. 8, 1986.
URL: http://www.gnu.org/bulletins/bull1.txt
(last visit 2012.02.07)

[7] The Free Software Definition Version 1.105.
URL: http://www.gnu.org/philosophy/free-sw.en.html
(last visit 2012.02.07)

[8] GNU General Public License v1.
URL: http://groups.google.com/group/gnu.announce/msg/bf254a45c6f512f3
(last visit 2012.02.07)

[9] A. G. González, "Viral contracts or unenforceable documents? Contractual validity of copyleft licences," *European Intellectual Property Review*, pp. 1-20, 2004.
URL: http://hdl.handle.net/1842/2263

[10] MIT License.
URL: http://www.opensource.org/licenses/MIT

[11] BSD License.
URL: http://www.opensource.org/licenses/bsd-license.php

[12] B. Perencs, "The Open Source Definition," *Open Sources: Voices from the Open Source Revolution*, 1999.
URL: http://www.oreilly.de/catalog/opensources/book/perens.html

[13] Open Source Initiative, "About", 2012.
URL: http://opensource.org/about

[14] Open Source Initiative, "The License Review Process", 2012.
URL: http://opensource.org/approval

[15] Why Open Source misses the point of Free Software.
URL: http://www.gnu.org/philosophy/open-source-misses-the-point.en.html
Last visit: 2012.03.18

[16] Why Copyleft?
URL: http://www.gnu.org/philosophy/why-copyleft.en.html
Last visit: 2012.03.18

[17] J. Lerner and J. Tirole, "The Scope of Open Source Licensing," *Source*, vol. 21, no. 1, 2005.
URL: http://jleo.oxfordjournals.org/content/21/1/20.short

[18] R. Sen, C. Subramaniam, and M. L. Nelson, "Determinants of the Choice of Open Source Software License," *Journal of Management Information Systems*, vol. 25, no. 3, pp. 207-240, Dec. 2008.

[19] J. Colazo and Y. Fang, "Impact of License Choice on Open Source Software Development Activity," *Journal of the American Society for Information Science*, vol. 60, no. 5, pp. 997-1011, 2009.

[20] M. Aslett, "The trend towards permissive licensing", 2011.
URL: http://blogs.the451group.com/opensource/2011/06/06/the-trend-towards-permissive-licensing/
Last visit: 2012.03.20

[21] M. Aslett, "FLOSSmole data confirms declining GPL usage", 2011.
URL: http://blogs.the451group.com/opensource/2011/06/13/flossmole-data-confirms-declining-gpl/
Last visit: 2012.03.20

[22] M. Aslett, "On the continuing decline of the GPL", 2011.
URL: http://blogs.the451group.com/opensource/2011/12/15/on-the-continuing-decline-of-the-gpl/
Last visit: 2012.03.20

[23] I. Herraiz, J. M. Gonzalez-barahona, G. Robles, U. Rey, and J. Carlos, "Towards a theoretical model for software growth ∗," *Fourth International Workshop on Mining Software Repositories (MSR'07)*, 2007.

[24] A. Deshpande, P. Alto, and D. Riehle, "The Total Growth of Open Source," *Source*, no. 2006, p. 3, 2008.

[25] G. Succi, J. Paulson, and A. Eberlein, “Preliminary Results from an Empirical Study on the Growth of Open Source and Commercial Software Products,” *EDSER-3 Workshop*, 2001.
URL: http://www.cs.virginia.edu/~sullivan/edser3/paulson.pdf

[26] M. Godfrey, “Evolution in open source software: A case study,” *Software Maintenance, 2000.*, 2000.
URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=883030

[27] M. Godfrey, “Growth, evolution, and structural change in open source software,” *workshop on principles of software evolution*, 2001.
URL: http://dl.acm.org/citation.cfm?id=602482

[28] C. Roy, “Evaluating the evolution of small scale open source software systems,” *See http://citeseer.ist.psu.edu/761885.html*, 2006.

[29] G. Robles and J. Amor, “Evolution and growth in large libre software projects,” *Proceedings of the Eighth International Workshop on Principles of Software Evolution*, 2005.
URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1572323

[30] C. Izurieta, “The Evolution of FreeBSD and Linux,” *Proceedings of the 2006 ACM/IEEE international*, 2006.

[31] R. Vasa, “Growth and Change Dynamics in Open Source Software Systems,” 2010.
URL: http://www.ict.swin.edu.au/personal/rvasa/helix/thesis/thesis.pdf

[32] S. Koch, “Evolution of Open Source Software Systems – A Large-Scale Investigation,” *International Conference on Open Source Systems*, 2005.
URL: http://oss2005.case.unibz.it/Papers/44.pdf

[33] S. Koch, “Software evolution in open source projects — a large-scale investigation,” *Journal of Software Maintenance and Evolution:*, no. June, pp. 361-382, 2007.
URL: http://onlinelibrary.wiley.com/doi/10.1002/smr.348/abstract

[34] S. Koch, “Evolution of Open Source Software Systems – A Large-Scale Investigation,” *International Conference on Open Source Systems*, 2005, p. 2.
URL: http://www.http://oss2005.case.unibz.it/Papers/44.pdf

[35] Ohloh, "The World's Oldest Source Code Repositories", 2007.
URL: http://www.ohloh.net/blog/worlds_oldest_source_code_repositories

[36] Ohloh API Reference. "ActivityFact,", 2007-
URL: https://www.ohloh.net/api/reference/activity_fact

[37] C. Ritz and J. C. Streibig, Nonlinear regression with R. Springer Verlag, 2008.

[38] L. Fahrmeir, Regression - Modelle, Methoden und Anwendungen. Springer Verlag, 2009.

[39] V. M. R. Muggeo, “segmented : an R package to fit regression models with broken-line relationships,” *R News*, vol. 8, no. 1, pp. 20-25, 2008.

URL: http://dssm.unipa.it/vmuggeo/segmentedRnews.pdf

[40] M. C. Newman, “Regression analysis of log-transformed data: Statistical bias and its correction,” *Environmental Toxicology and Chemistry*, vol. 12, no. 6, pp. 1129–1133, 1993. URL: http://onlinelibrary.wiley.com/doi/10.1002/etc.5620120618/abstract

[41] K. Burnham, “Model selection and multimodel inference: a practical information-theoretic approach,” vol. 252, pp. 267-351, 2002.

License