

# The State and Development of Gender Equality in Computer Science Publications

Bachelor Thesis

Linda Fernsel  
s0555949

July 24, 2019

**Supervisor** Prof. Dr. Gefei Zhang  
**First Examiner** Prof. Dr. Gefei Zhang  
**Second Examiner** Prof. Dr. Debora Weber-Wulff

# Abstract

How big is the Gender Gap when it comes to authors of publications in the field of computer science? The purpose of this thesis is to answer this question considering aspects like development over time and whether a publication was created by a team of authors. These aspects are investigated by a bibliometric analysis of publication data obtained from the *dblp computer science bibliography* in combination with *NamSor API v2* which derives the gender of an author name.

# Acknowledgments

I would like to thank my supervisor Prof. Dr. Gefei Zhang from the University of Applied Sciences Berlin for his time, patience and good council and of course for the opportunity. A huge thank you goes to Elian Carsenat of *NamSor* who supported my work with API credit points and answers to my thousand questions. Thanks also to Prof. Dr. Wilfried Elmenreich of Alpen-Adria University Klagenfurt for his helpful feedback. I have Dr. Martin Bretschneider to thank for the *LaTeX* template I used for the layout of this thesis. The template is available for free at <https://www.bretschneider.net.de/tips/arbeit-vorlagen.zip>.

Linda Fernsel, July 24, 2019

# List of Figures

2.1	Histogram of score of abbreviated names . . . . .	12
2.2	Histogram of score of abbreviated names with a male Russian last name . . .	12
2.3	Histogram of score of names with common Asian family names . . . . .	13
2.4	Histogram of score of names with an unisex first name . . . . .	13
2.5	Histogram of score by gender before data improvements . . . . .	14
2.6	Histogram of score by gender after data improvements . . . . .	16
3.1	How many male and female authors are there in the <i>dblp</i> data set? . . . . .	21
3.2	Histogram of the Gender Balance Indicator of publications written in a team .	25
3.3	Scatter graph to explore the question whether bigger teams are more diverse	27
3.4	Share of female authors and their contributions per year . . . . .	32
3.5	Share of female and unclassified authors per year . . . . .	32

# List of Tables

2.1	Information returned by NamSor API v2 . . . . .	7
2.2	Score, accuracy and distribution for names tested by <i>NamSor</i> . . . . .	8
2.3	Structure and content of the data set of publications . . . . .	17
3.1	Mean, median and mode of the amount of contributions made by male and female authors . . . . .	29

# Contents

List of Figures	IV
List of Tables	V
1 Introduction	1
2 Prerequisites	3
2.1 Technologies . . . . .	3
2.1.1 Python and Python libraries . . . . .	3
2.1.2 NamSor . . . . .	4
2.2 Data available for research . . . . .	4
2.2.1 Publications . . . . .	4
2.2.2 Gender . . . . .	6
2.3 Preparing data for analysis . . . . .	9
2.3.1 Extracting author names . . . . .	9
2.3.2 Classifying names by gender . . . . .	10
2.3.3 With which score is a name reliably classified? . . . . .	11
2.3.4 Improving classification . . . . .	14
2.3.5 Structuring bibliometric data . . . . .	17
3 Analysis	19
3.1 Are there as many female as male authors in the field of computer science that could be role models? . . . . .	19
3.1.1 Hypothesis . . . . .	19
3.1.2 Condition . . . . .	19
3.1.3 Method . . . . .	19
3.1.4 Result . . . . .	20
3.1.5 Threads to validity: Approximation of the amount of male and female authors in the field of computer science . . . . .	21

## Contents

3.2	Are groups of collaborating authors diverse, setting a good example for other areas of life? . . . . .	22
3.2.1	Hypothesis . . . . .	22
3.2.2	Condition . . . . .	22
3.2.3	Method . . . . .	23
3.2.4	Result . . . . .	24
3.2.5	Threads to validity: Approximation of the amount of men and women by team . . . . .	24
3.3	Is the amount of co-authors correlated to the diversity of a team? . . . . .	25
3.3.1	Hypothesis . . . . .	25
3.3.2	Condition . . . . .	25
3.3.3	Method . . . . .	26
3.3.4	Result . . . . .	26
3.3.5	Threads to validity: Approximation of the amount of men and women by team . . . . .	27
3.4	Are female authors as productive as male authors? . . . . .	27
3.4.1	Hypothesis . . . . .	27
3.4.2	Condition . . . . .	28
3.4.3	Method . . . . .	28
3.4.4	Result . . . . .	29
3.4.5	Threads to validity: Approximation of the amount of productivity per author . . . . .	29
3.5	Did gender equality improve over time and can therefore an increase in the amount of role models be expected? . . . . .	30
3.5.1	Condition . . . . .	30
3.5.2	Method . . . . .	30
3.5.3	Result . . . . .	31
3.5.4	Threads to validity: Approximation of the amount of male and female authors by year . . . . .	31
4	Conclusion and future work	33
	Bibliography	35

# 1 Introduction

According to *The Organisation for Economic Co-operation and Development* (OECD) the Gender Gap in scientific publication authorship within several different fields is rather big and closing slower than in other careers (OECD 2016). According to (Google 2014, p. 8), the Gender Gap in computer science might be caused by factors anchored in society. Young women, for instance, often lack encouragement by their parents and friends to pursue a career in computer science and they perceive such careers negatively. Role models, encouraging young women and changing their view on what a computer scientist is and does, could help encounter the Gender Gap. Some of those role models could be women publishing in the field of computer science. The purpose of this thesis is to answer the following questions:

- Are there as many female as male authors in the field of computer science that could be role models?
- Are groups of collaborating authors diverse, setting a good example for other areas of life?
- Is the number of co-authors correlated to the diversity of a team?
- Are female authors as productive as male authors?
- Did gender equality improve over time and can an increase in the amount of role models therefore be expected? Maybe it is just a matter of time until the Gender Gap closes without further efforts being needed?

The approach chosen in this thesis for answering these questions concerning the current state of gender equality, its development and the influence of the size of a team of co-authors, relies on data scientific methods. They include statistical hypothesis tests on bibliometric data taken from the *dblp computer science bibliography*<sup>1</sup> combined with *NamSor API v2*<sup>2</sup> to infer the gender of an author from their name.

---

1 <https://dblp.org/>

2 <https://v2.namsor.com/>



## *1 Introduction*

This thesis is organized as follows. Chapter 1 introduces by formulating research goals and explaining how they are approached. Section 2.1 presents the most important technologies used for this work. Chapter 2 Section 2.2 introduces data available for analysis and evaluates its usefulness. Preparation of this data including structuring it, linking data, improving its quality and calculating additional variables is described in Section 2.3. In Chapter 3, hypotheses are formulated. Each is tested statistically and its result interpreted. Validity constraints are discussed as well. Finally Chapter 4 draws conclusions and suggests what future work could follow this thesis.

# 2 Prerequisites

## 2.1 Technologies

The following section describes which technologies are mainly used for preparing and executing the bibliometric analysis.

### 2.1.1 Python and Python libraries

All analysis is conducted using the programming language *Python 3.7.1* with *Jupyter Notebook 5.7.4*<sup>1</sup> as an Interface<sup>2</sup>. *Python* is used for its good documentation and known flexibility for data analysis, an attribute owed to its many libraries. Of those libraries the following are especially notable:

- *ElementTree XML API 1.3.0*<sup>3</sup> provides a tree structure for parsing, storing and navigating XML files
- *Pandas 0.23.4*<sup>4</sup> makes CSV data easily usable, especially when selecting and grouping data and applying calculations to data
- *SciPy 1.1.0*<sup>5</sup> with *Numpy 1.15.4*<sup>6</sup> helps perform statistical tests on large sets of data
- *Matplotlib 3.0.2*<sup>7</sup> visualizes data

---

1 included in *Anaconda 3*: <https://www.anaconda.com/>

2 The *Jupyter Notebooks* programmed and used for this thesis have been made open source by the author of this thesis and can be found at <https://github.com/LiFaytheGoblin/Gender-Equality-in-CS-Publications>.

3 <https://docs.python.org/3.7/library/xml.etree.elementtree.html>

4 <https://pandas.pydata.org/>

5 <https://www.scipy.org/>

6 <https://www.numpy.org/>

7 <https://matplotlib.org/>

## 2 Prerequisites

### 2.1.2 NamSor

*NamSor API 2.0.2-beta*<sup>8</sup> is used to classify names by gender and to parse full names into first and last names. As an intermediary between *Python* and the *NamSor API*, *NamSor Python SDK*<sup>9</sup> is used. *NamSor Python SDK* is released under the *GNU Affero General Public License v3.0* (NamSor 2019b). The SDK can be downloaded from *GitHub* where it is continuously updated.

## 2.2 Data available for research

This section describes the data available for analysis, its structure, its quality and quantity.

### 2.2.1 Publications

The *dblp computer science bibliography (dblp)*<sup>10</sup> aims to provide a bibliography of the most relevant publications in the field of computer science written mostly in English language and aimed at an international audience (dblp computer science bibliography 2019g). *dblp* is a service of the non-profit company *Schloss Dagstuhl - Leibniz-Zentrum for Informatics* in cooperation with the *University of Trier*, Germany (dblp computer science bibliography 2019e).

The data on publications is provided under the *Open Data Commons ODC-BY 1.0* license, allowing anyone to “copy, distribute, use, modify, transform, build upon and produce derived works” (Open Knowledge Foundation 2018) if a reference to *dblp* is given (dblp computer science bibliography 2019c). The data is being updated daily but a versioned release allowing for scientific reproduction of an analysis is created monthly and can be downloaded as a zip-file at <https://dblp.org/xml/release/> (dblp computer science bibliography 2019a). For the analysis in this thesis the data release of April 1<sup>st</sup> 2019 has been used. It contains about 4,5 million computer science publications written by over 2 million authors.

The data provided by *dblp* comes in semi-structured form. The format is XML, encoded in *ISO-8859-1*, also known as *latin-1* (dblp computer science bibliography 2019d). The structure

---

<sup>8</sup> <https://v2.namsor.com/>

<sup>9</sup> <https://github.com/namsor/namsor-python-sdk2>

<sup>10</sup> <https://dblp.org/>

## 2 Prerequisites

of the XML is much like a list, containing elements that represent either a publication or a person. Person records are not nested in publication records as one might expect. Instead they are hierarchical siblings to publication records. Each publication record contained in the XML file is given a key, has a title, a publication year and one or more author entries. Additional information may be given. Most publications are conference and workshop papers or journal articles, but there are also books, chapters, PhD and master theses and web pages (dblp computer science bibliography 2019b). A web page record, tagged with “<www>”, may be a web page or be used as a person record, containing the names, web page and affiliations of an author (dblp computer science bibliography 2019f).

The following two examples of XML elements have been taken from (dblp computer science bibliography 2019d). The first example is a publication entry. It contains elements describing information like this article’s author, title or a URL for online access.

```
<article key="journals/cacm/Gentry10" mdate="2010-04-26">
  <author>Craig Gentry</author>
  <title>Computing arbitrary functions of encrypted data.</title>
  <pages>97-105</pages>
  <year>2010</year>
  <volume>53</volume>
  <journal>Commun. ACM</journal>
  <number>3</number>
  <ee>http://doi.acm.org/10.1145/1666420.1666444</ee>
  <url>db/journals/cacm/cacm53.html#Gentry10</url>
</article>
```

The second example is a “<www>” element representing a person record. It contains further elements that represent this author’s name and home page URL.

```
<www mdate="2004-03-23" key="homepages/g/OdedGoldreich">
  <author>Oded Goldreich</author>
  <title>Home Page</title>
  <url>http://www.wisdom.weizmann.ac.il/~oded/</url>
</www>
```

## 2 Prerequisites

### 2.2.2 Gender

The data on publications obtained from *dblp* does not disclose information on the gender of an author. Thus, the gender needs to be inferred from an author's name. This method can only make an estimation from the name, it can not determine how closely an author feels associated with a gender. The method only differentiates between the gender classes "male" and "female". While these constraints evidently lead to imprecision, this method is deemed sufficient for the cause of this thesis.

The gender of an author is inferred using *NamSor API 2.0.2-beta*<sup>11</sup> with the *NamSor Python SDK*<sup>12</sup>. *NamSor* is a French company that has provided its service for gender studies related research, including (Mattauch, Lohmann, Hannig, Lohmann & Teich 2018) and (*She Figures* 2015). In (Mattauch et al. 2018), *NamSor v1* was combined with *dblp*, researching on the diversity of authors of specific conference papers of the years 2012 to 2014. This thesis, on the other hand, uses *NamSor v2*, looks at a longer period of time, asks questions about not only the share of female authors but about productivity, teams and timely development and takes all kinds of publications into account when answering those questions. The *NamSor API* is well suited for research on gender because it considers a diverse range of names, considers cultural influence on which gender is linked and also informs about the likeliness of the assumption.

The API has different endpoints for guessing a gender. Some endpoints use the first and last name, some a full name, some additionally take geographical context. All API endpoints require an access key and use up API credit points.

Before a call to the API can be made via the SDK, the SDK first has to be imported into *Python*, then the developer needs to authorize the SDK using their API key and finally an instance of the *PersonalApi* class has to be created. The *PersonalApi* class contains functions that can be used to call API endpoints that classify names. The following code shows how the SDK is set up for use. It was taken from (NamSor 2019c) and adapted.

```
# Import the SDK into Python
import openapi_client

# Configure the API authorization by providing an API key
configuration = openapi_client.Configuration()
configuration.api_key['X-API-KEY'] = 'EXAMPLEKEY'
```

---

<sup>11</sup> <https://v2.namsor.com/>

<sup>12</sup> <https://github.com/namsor/namsor-python-sdk2>

## 2 Prerequisites

```
# Create an instance of the PersonalAPI class
pers_api_instance = openapi_client.PersonalApi(
    openapi_client.ApiClient(configuration))
```

The data on the presumable gender of a name is returned by the *NamSor Python SDK* in the form presented in Table 2.1 (NamSor 2019a).

key	Type	description
id	String	
name	String	The tested name
likelyGender	String	The probable gender: “male” or “female”
genderScale	Int	The gender expressed as -1 or 1: “-1” means “male”, “1” means “female”. Available for backwards compatibility.
score	Double	The likeliness of a correct result, a number > 0

**Table 2.1:** Information returned by NamSor API v2

For example when calling the API endpoint for classifying the full name “Emily Brontë” with the SDK function `pers_api_instance.gender_full('Emily Brontë')`, the following object is returned:

```
{
  'id': None,
  'name': 'Emily Brontë',
  'likely_gender': 'female',
  'gender_scale': 1.0,
  'score': 14.29510518198754
}
```

Names can also be classified in batches with a maximum size of 1000 names, with the method `pers_api_instance.gender_full_batch()`. When classified in batches each name must be given an id by the programmer. The id is later returned as part of the result object. If a name is not classified as a batch, no id is entered and no id is returned.

**The meaning of “score”** The score returned for a name aims to describe the quality of the result. However, it does not allow to calculate how probable it is that the name was classified incorrectly and is not comparable between different API endpoints and different API versions

## 2 Prerequisites

(2.0.2-*beta* and 2.0.5 for instance). This is a drawback that *NamSor* solved in version 2.0.5<sup>13</sup>. Version 2.0.5 was published mid July 2019, after the experimental work for this thesis had been completed.

*NamSor* tested the API with a set of diverse names of inhabitants of Boston (United States)<sup>14</sup>. This test makes the meaning of the score value clearer. Table 2.2 displays the accuracy and distribution value for names classified with at least a certain score, as obtained by *NamSor*. The accuracy answers the question “Out of all names classified with at least a certain score, how many have been classified correctly?”. The distribution value says how many names have been classified with at least a certain score.

score (rounded)	accuracy	distribution
0	95%	100%
1	96%	98%
2	97%	93%
3	97%	86%
4	98%	73%
5	99%	56%
6	99%	40%
7	99%	25%

**Table 2.2:** Score and corresponding accuracy and distribution for the tested classification of names of inhabitants of Boston

The table shows that all names are classified with an accuracy of 95%, meaning that 95% of all names are classified correctly. 98% of all names have a score of at least 1. 95% of these names are classified correctly. 56% of all names have a score  $\geq 5$ . 99% of them have been classified correctly. The accuracy does not improve for a minimum score bigger than 5, but less names are classified with these scores.

All in all the result quality “depends on the sample name distribution (% European, African, Asian names ...) as well as the input’s data quality”<sup>15</sup>. That means that the accuracy and distribution displayed above are exact for the specific set of names used but only an approximation for a diverse set of names. Accuracy is higher for a set of only European names, as shown by another test conducted by *NamSor* on version 1 of the API (note that *NamSor* commonly calls accuracy “precision” and the distribution value “recall”) (NamSor 2018). With a data set such as the one from *dblp*, the degree of ethnic diversity is expected to be very high, considering it collects mainly international publications (*dblp* computer science bibliography 2019g). Thus, the accuracy is expected to be lower or equally low as the accuracy of the tested names of

---

<sup>13</sup> Personal correspondence with *NamSor*

<sup>14</sup> Personal correspondence with *NamSor*

<sup>15</sup> Personal correspondence with *NamSor*

## 2 Prerequisites

inhabitants of Boston. However, the exact probability of the resulting classifications of the author names being correct can not be determined reliably since this would require verifying the classifications of a part of the data by hand which is beyond the scope of this work and will still result in authors whose gender can not be determined within a sensible amount of time.

### 2.3 Preparing data for analysis

This section documents how the available data is transformed into useful structures and how data relevant to analysis is extracted and calculated.

#### 2.3.1 Extracting author names

The raw data on publications from *dblp* is first parsed to an *ElementTree*, using the *ElementTree XML API*. Because the XML file obtained from *dblp* only contains ASCII characters but encodes characters from *ISO/IEC 8859-1* (dblp computer science bibliography 2019d), additional key-value-pairs need to be added to the XML parser. An example for such a key-value-pair would be ('ccedil', 'ç') with 'ccedil' being the key and 'ç' being the corresponding value. By adding this key-value-pair to the XML parser, the parser can later recognize the series of characters ;ccedil in the XML file and parse it as ç. All in all, 65 key-value-pairs for *ISO/IEC 8859-1* characters are added to the parser in order for it to parse the XML properly to an *ElementTree*.

The resulting *ElementTree* contains 6804878 entries. Most of the 2260493 "<www>" entries are person records and some are web pages. They are not needed and removed. This leaves 4544385 entries of publications left to analyze.

From the resulting tree, author names are extracted with the following code. Fields for their gender classification and *NamSor* score are prepared, receiving default values for now.

```
# Get the root of the ElementTree containing all publication entries
root = tree.getroot()

# Python dictionary for all author names
names_complete = {}
```



## 2 Prerequisites

```
for publication in root:
    # Get author elements of this publication
    authors = publication.findall('author')

    # Get names from author elements and add them to the dictionary of names
    for author in authors:
        name = author.text
        if (name not in names_complete):
            names_complete[name] = {
                'name': name,
                'likely_gender': 'not determined',
                'score': 0
            }

# Convert dictionary to Pandas Dataframe
names_df = pandas.DataFrame.from_dict(
    names_complete, orient='index', dtype=None)
```

For the analysis, an uncleaned set of 2306418 individual author names is available.

### 2.3.2 Classifying names by gender

For determining the probable gender of a name the *NamSor* API endpoint for classifying full names batch-wise without any additional information is used. The following code is used for making the API calls. It has been shortened where appropriate.

```
# Get a list of all author names to be classified
names_stack = list(names_df.index.values)

batch_size = 1000
result = []

def create_batch(names):
    # Format a list of names into a batch the SDK can use
    # ...
```

## 2 Prerequisites

```
while (len(names_stack) >= batch_size):
    try:
        # Make the API call and add the returned object to the result list
        result.append(pers_api_instance.gender_full_batch(
            batch_personal_name_in=create_batch(names_stack[0:batch_size])))

        # Delete what has already been classified
        del names_stack[0:batch_size]

        # Classify remaining names if there are less names than a batch size
        if(len(names_stack) < batch_size and len(names_stack) > 0):
            result.append(pers_api_instance.gender_full_batch(
                batch_personal_name_in=create_batch(names_stack)))
            names_stack = []
    except ApiException as e:
        # Handle exceptions such as an interrupted connection
        # ...

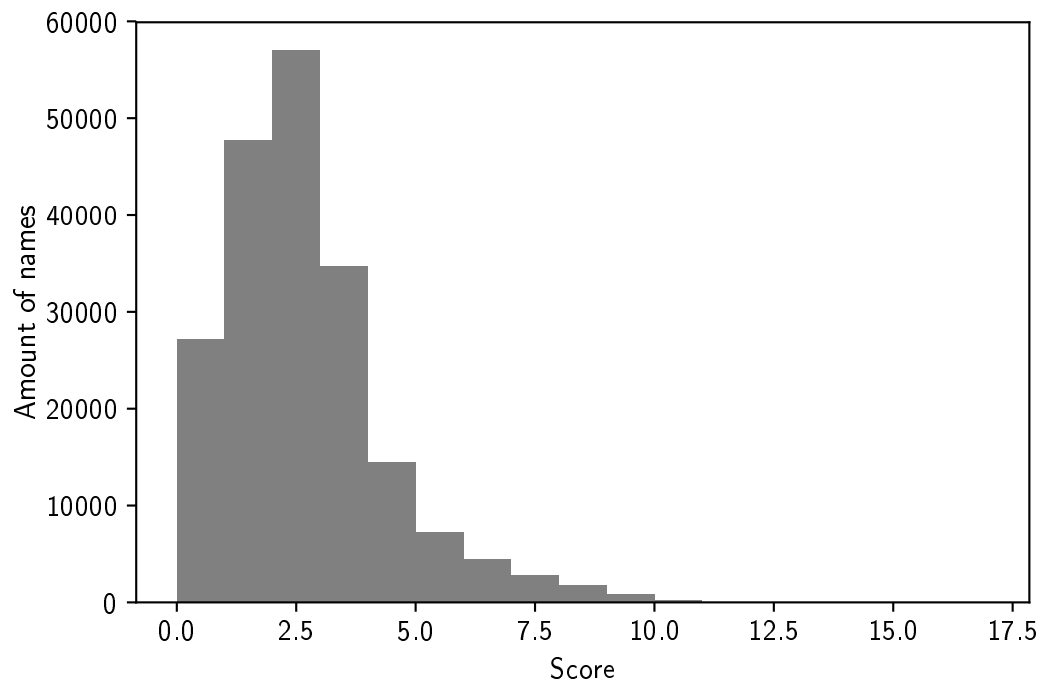
# Save results into the Pandas Dataframe of names
for oapi_el in result:
    names_df.at[oapi_el.id, 'likely_gender'] = oapi_el.likely_gender
    names_df.at[oapi_el.id, 'score'] = oapi_el.score
```

### 2.3.3 With which score is a name reliably classified?

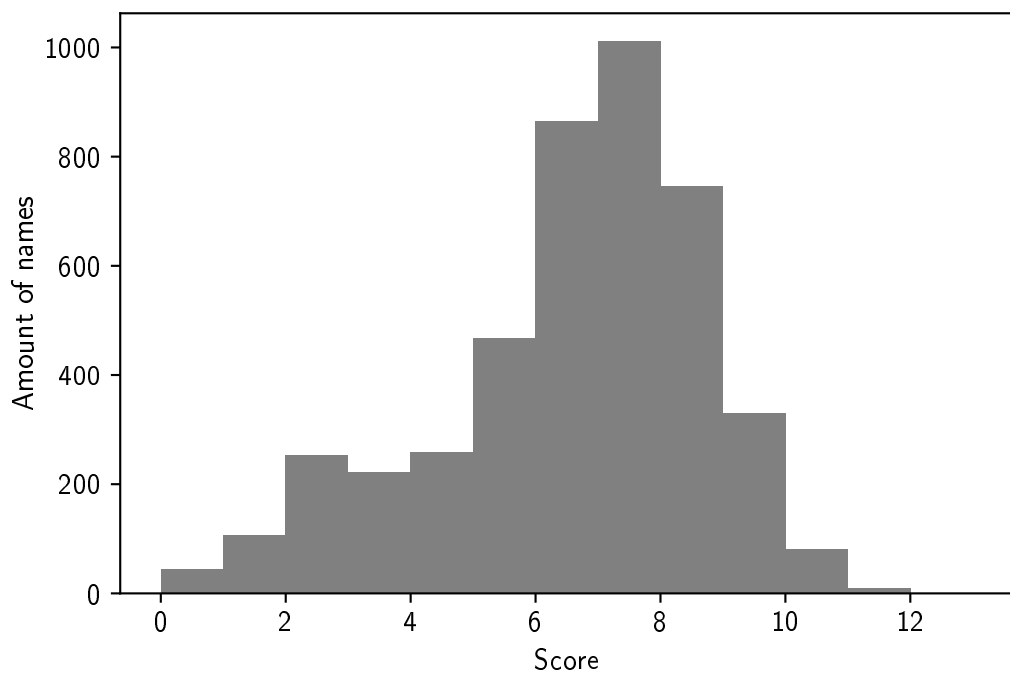
There is no way to exactly determine the score for which a result is most probably correct. However efforts can be made to estimate the minimum reliable score based on which scores are accorded to difficult groups of names. The distributions of names containing (a) abbreviated names, (b) male Russian family names ending in “ov”, “ev”, “vich” or “sky”, (c) common Asian family names like “Chen” or “Liu” and (d) the unisex names “Kim”, “Ashley” or “Anh” are examined. Their histograms are shown in Figures 2.1 to 2.4. Group (a) has a median and a mode score of 2. Group (b) has a median score of 6 and a mode of 7, since the gender can be inferred from the ending or from a second, non abbreviated first name. Group (c) names are classified with a median and mode score of 2. Names from group (d) are given a median and mode score of 1.

For abbreviated names the score of 5 marks which names could be classified and which could not be classified: The distribution of Russian classifiable names rises at 5 while of all names

## 2 Prerequisites

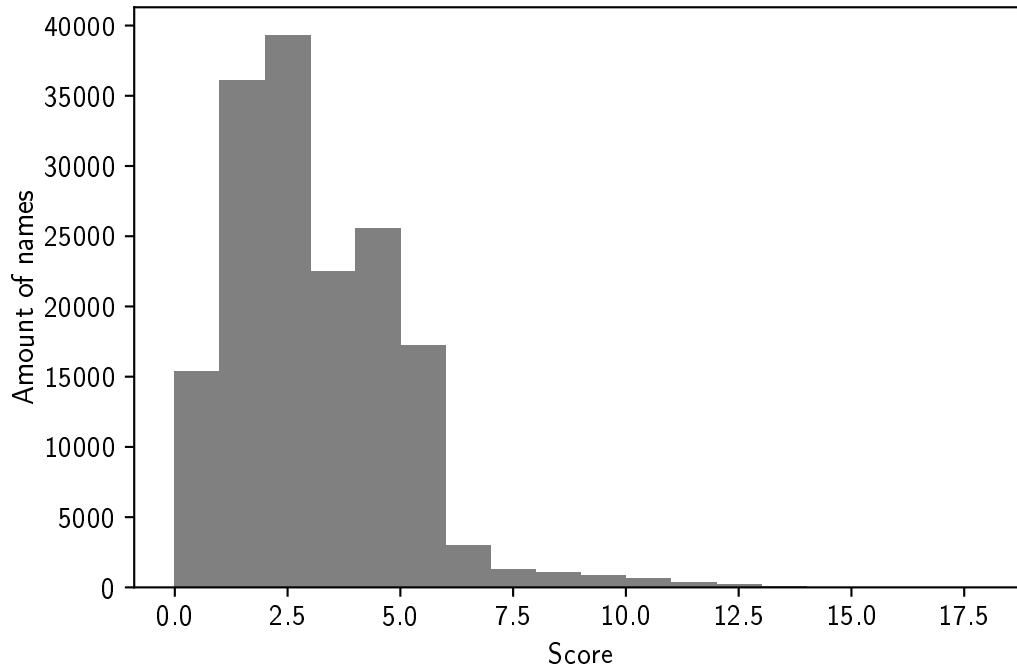


**Figure 2.1:** Histogram of score of abbreviated names

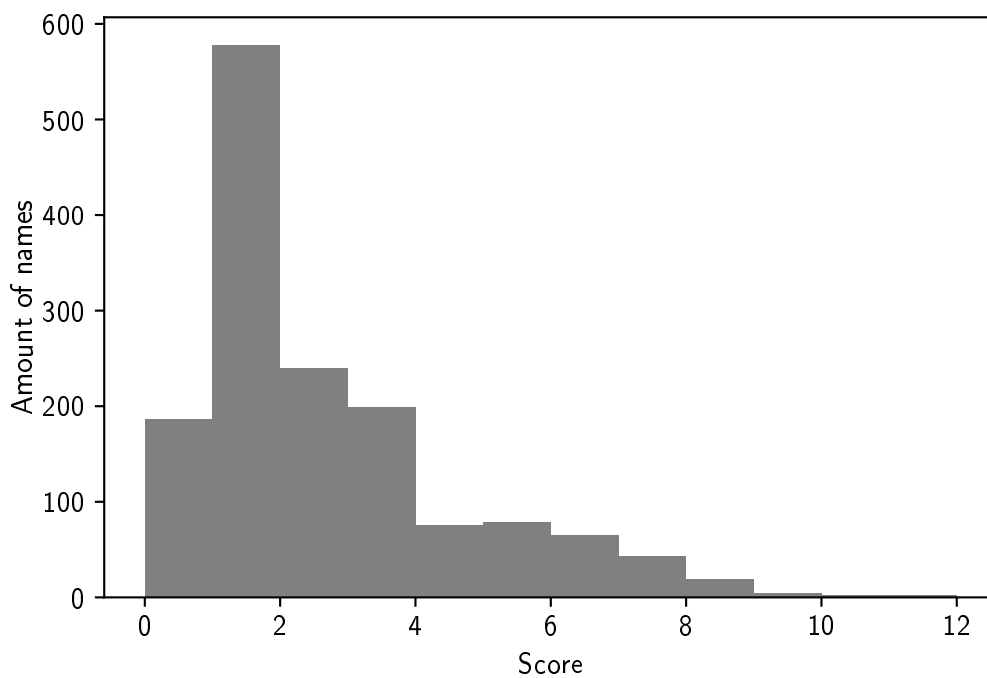


**Figure 2.2:** Histogram of score of abbreviated names with a male Russian last name

## 2 Prerequisites



**Figure 2.3:** Histogram of score of names with common Asian family names



**Figure 2.4:** Histogram of score of names with an unisex first name

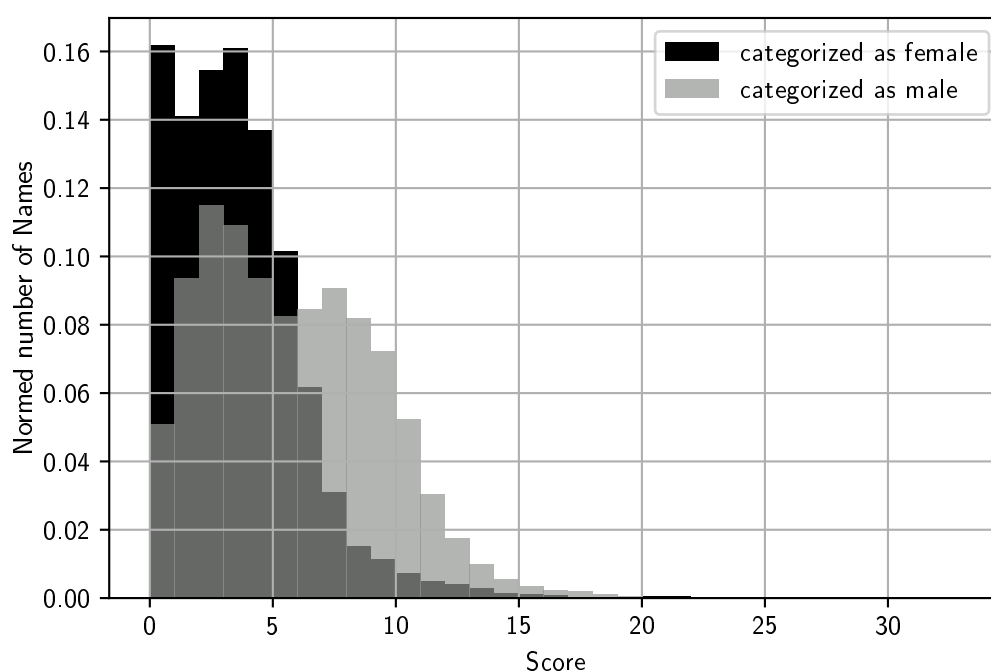
## 2 Prerequisites

with abbreviated names only few have a score of 5 or higher. For names with Asian family names the distribution decreases at 6. However it should be noted that many names with Asian family names have English first names which can be classified. For unisex names the score distribution decreases at 4.

It can be estimated that a score of 5 or more is a good enough indicator for a name to be classified reliably, therefore only names with a score of 5 or higher will be taken into consideration for analysis.

1127184 of 2306418 names (48.87%) have a score of 5 or higher and are therefore considered to be classified correctly.

### 2.3.4 Improving classification



**Figure 2.5:** Histogram of score before data improvements. The y axis represents the normed amount of names with a certain score of a certain gender. “Normed” means that that the amount is the relative amount of names out of all names of this gender, making both gender categories comparable. The medium grey that is not contained in the legend is where the amounts of names for both gender classes intersect.

The score values by gender are not equally distributed, as can be seen in Figure 2.5. More reliable results could not only provide more data available for analysis but also a more equal

## 2 Prerequisites

distribution of the score value by gender. A different method of classification might therefore be necessary.

In the following, the most precise method is explained and considered. Then, a compromise is described and evaluated.

In the most precise method, to be called method P, all authors are first extracted from "<www>" elements by looking for a key containing "homepages/". The aliases used by an author available in a "<www>" element are to be marked as "abbreviated" or "full name". All full names are cleaned. That means that they are stripped from numbers indicating multiple authors with the same name and other characters such as parentheses. The parsed names are then sent to *NamSor* again, this time to guess a geographical context. With this geographical context the parsed names are classified by *NamSor* because the endpoint taking parsed first and last names and geographical context is supposed to return the best results<sup>16</sup>. When there are multiple classified names per author the classification with the highest score is used for further analysis. All other authors found in the publication data that were not included as a "<www>" element are also either first cleaned and parsed into first and last names if it the name is a full name or directly put in geographical context and finally classified.

The advantages of this method are that (a) the amount of individual authors may be approximated more precisely, (b) the "<www>" may contain full name aliases so abbreviated names can still be classified, (c) the "<www>" element sometimes contains a unicode name for Asian names that may contain information on gender (Mattauch et al. 2018, p. 4) - therefore *NamSor* can classify them more reliably than their latin version, (d) information on geographical context is retrieved which could be used to make observations based on country. However the disadvantages are that (e) *dblp* sometimes merges multiple authors into one "<www>" element by mistake, this could neutralize advantage (a), (f) as there are only about 1.02 times as many names as authors the effect of (b) and (c) can be expected to be rather small and (g) the request for geographical context takes 10 times as many API credit points than a call for parsing or classification which makes this method way more expensive than the one that was previously used in this thesis. It can thus be concluded that this alternative method is more precise but not reasonable in this case.

The following method is a compromise, to be called method C, that tries to decrease the number of unclassifiable authors while not using geographical context and the expensive API usage that comes with it. This method was first tested on samples of different scores. They showed that method C is most effective for redistributing the resulting scores more

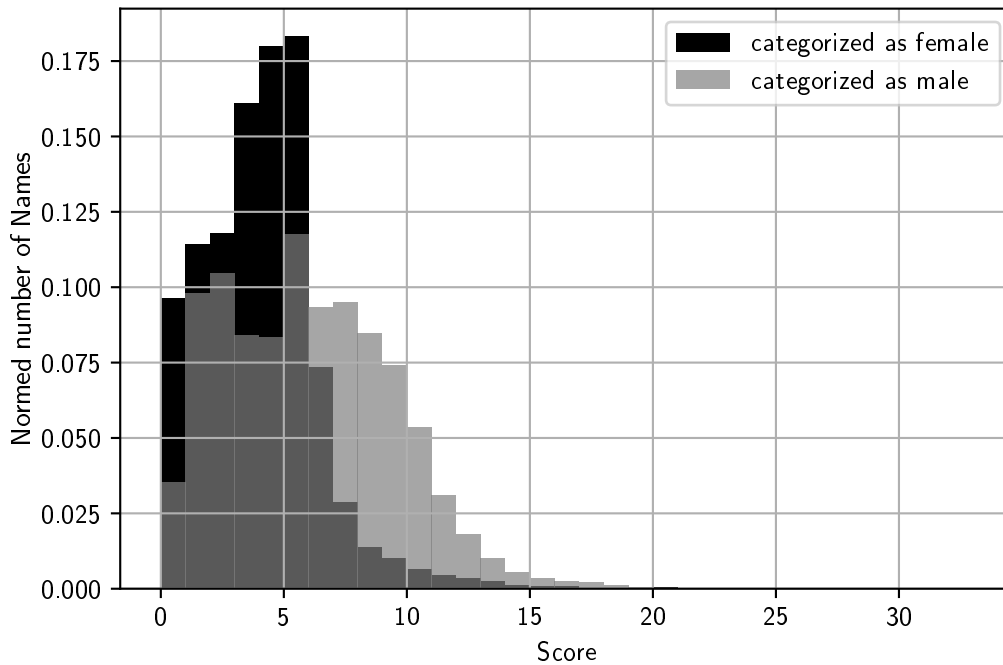
---

<sup>16</sup> Personal correspondence with *NamSor*

## 2 Prerequisites

evenly when performed on names that have priorly been marked with a low score (0 to 5).

Since 5 has been determined to be the minimum reliable score, names that have already been evaluated by *NamSor* and received a classification with a score smaller than 5 are considered for reclassification. The full name is cleaned as described in method P and then parsed into first names and last names by *NamSor*. The parsed names are then classified again using the API endpoint that takes first and last names. This API endpoint is less reliable than the one additionally taking geographical context but it is still more reliable than the endpoint taking a full name<sup>17</sup>.



**Figure 2.6:** Histogram of score after data improvements. The amount of names with a certain score is normed and not the absolute amount of names. The medium grey is where both data for both gender classes intersect.

After the reclassification 54.47% of all names are considered to be reliably classified, which is 129131 names more than before. The distribution of the score values by gender is slightly more equal as can be seen in Figure 2.6. It can be concluded that with a reasonable amount of effort and cost, this is the best possible result.

<sup>17</sup> Personal correspondence with *NamSor*

### 2.3.5 Structuring bibliometric data

The next goal is to convert the semi-structured XML data on publications into structured table form, adding information on gender distribution per publication. The resulting form and content of the publications table is displayed in Table 2.3.

Column name	Type	Description	Example
key	String		"books/acm/kim95/ AnnevelinkACFHK95"
title	String		"Object SQL - A Language for the Design and Implementation of Object Databases."
pub_cat	String	Category, can be a "book", "journal", "article" ...	"incollection"
pub_type	String	Type, can be a "survey", "data", "informal", ...	None
year	String		1995
authors	String[]		["Jurgen Annevelink", "Rafiul Ahad", "Amelia Carlson", "Daniel H. Fishman", "Michael L. Heytens", "William Kent"]
n_authors	Int	Amount of authors	6
n_males	Int	Amount of men	4
n_females	Int	Amount of women	1
n_unknown	Int	Amount of non- classifiable authors (score < 5)	1
ratio	Float	an indicator for gender balance from -50% to 50%	30,0
n_contribs_females	Float	Part contributed by women	0.166667
n_contribs_males	Float	Part contributed by men	0.666667
n_contribs_unknown	Float	Part contributed by non-classifiable authors	0.166667

**Table 2.3:** Structure and content of the data set of publications



## 2 Prerequisites

**The meaning of “ratio”** The value called “ratio” in the table is an indicator for the balance of male and female authors. It signifies the amount of male authors that would need to be replaced by female authors in order to establish gender equality. This does not need to take into account authors of unknown gender. Let this value be called “Gender Balance Indicator”. The Gender Balance Indicator of a publication in %  $r$  is calculated from the total amount of male and female names  $n$  and the amount of female names  $n_{female}$ :

$$r = \frac{\frac{n}{2} - n_{female}}{n} \times 100$$

For example, if there are 2 male authors and 0 female authors the Gender Balance Indicator is 50% because 50% of all male authors would need to be replaced by women to obtain a balanced authorship. If there is 1 male and 1 female author, the Gender Balance Indicator is 0. If there are 1 male and 3 female authors, the Gender Balance Indicator is -25% since one female author would need to be replaced by a male author for a balanced authorship.

## 3 Analysis

In the analysis one hypothesis and one alternative hypothesis are formulated for each of the five questions raised in Chapter 1 of this thesis. Then, a method of testing is described for each hypothesis and the results are interpreted.

### 3.1 Are there as many female as male authors in the field of computer science that could be role models?

#### 3.1.1 Hypothesis

$h_0$  = There are as many men as women who publish in the area of computer science.

$h_1$  = There are more or less men than women who publish in the area of computer science.

#### 3.1.2 Condition

The hypothesis  $h_0$  is true if the observed distribution of authors by gender is not significantly different than the expected gender distribution.

#### 3.1.3 Method

The Chi Square Test ( $\chi^2$  Test) is usually used to test whether two samples have been taken from the same set of data (Pearson 1911, p. 250). For that, the expected distribution of observations  $F$  is calculated and compared with the actual distribution  $O$ . If the difference is big enough it can be assumed that the samples have not been taken from the same set of data. Since part of the  $\chi^2$  Test is to compare observed frequencies  $O$  to expected frequencies  $F$  it can be used to verify if the observed distribution of male and female authors  $O$  is significantly

### 3 Analysis

different than the expected  $F$ . All assumptions for the  $\chi^2$  Test are met by the available data (See also (Pearson 1911, p. 250)):

- The variable of interest, the gender of an author, is nominal (“male” or “female”) and thus the frequency  $O$  for each value can be counted
- Observations are independent, since single authors are independent from each other

In order to analyze the distribution of male and female authors  $O$ , the amount of male and female authors needs to be estimated. The amount of male or female authors is approximated with the amount of male or female names extracted from the *dblp* data, considering only those names that have been classified reliably, that is with a score equal or higher than 5. 1256315 names can be used for analysis. The data set reveals  $O$  to be:

$$O(\text{male authors}) = 1112117$$

$$O(\text{female authors}) = 144198$$

The expected ratio of male to female authors is 1:1. The expected distribution  $F$  is therefore half of all available names for each gender:

$$F(\text{male authors}) = 628157.5$$

$$F(\text{female authors}) = 628157.5$$

The  $\chi^2$  Test is executed using the method `stats.chisquare` from *Python's SciPy* library. The method takes the array of the observed distribution  $O$  as a first parameter and the array containing the expected distribution  $F$  as a second parameter. It returns the test statistic  $\chi^2$  and the two-sided p-value  $p$  (The Scipy community 2018a). The p-value  $p$  is the probability of obtaining the observed results under the condition that the hypothesis  $h_0$  is true. If the two-sided p-value  $p$  is smaller than the chosen level of significance  $\alpha$  it indicates that  $H_0$  is to be rejected. Otherwise  $h_0$  can be accepted.

#### 3.1.4 Result

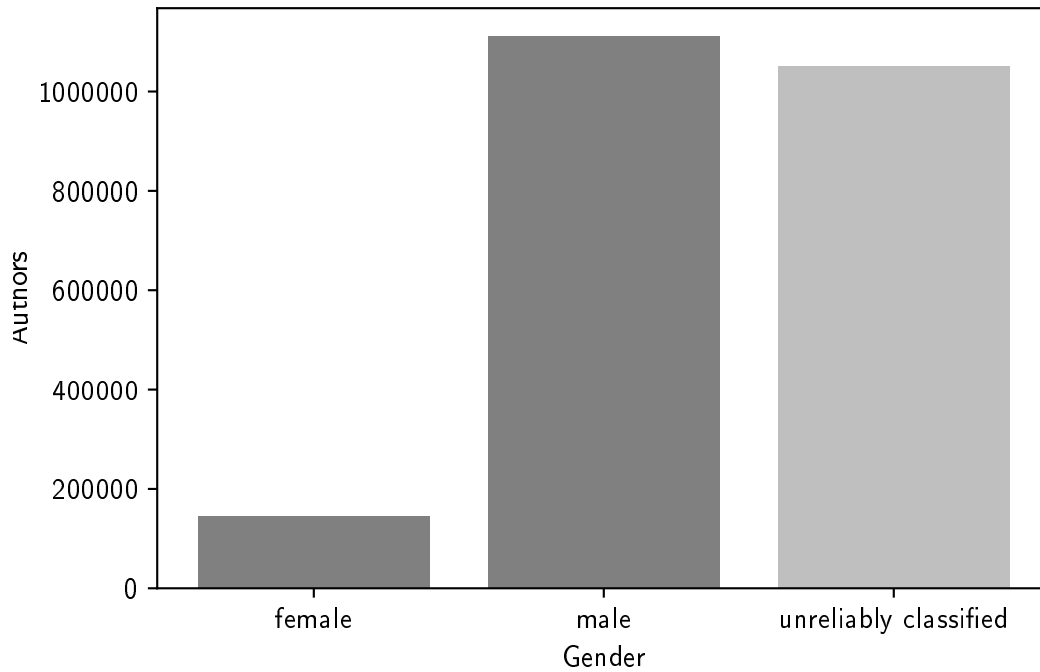
$$\chi^2(O, F) \approx 745726$$

$$p = 0$$

$$\alpha = 0.05$$

The hypothesis  $h_0$  can be rejected because  $p$  is smaller than  $\alpha$ . This means that there are more or less men than women who publish in the area of computer science. Figure 3.1 shows that there are more male than female authors. Women can find less role models to identify with in the group of computer science authors than men.

### 3 Analysis



**Figure 3.1:** How many male and female authors are there in the *dblp* data set?

#### 3.1.5 Threads to validity: Approximation of the amount of male and female authors in the field of computer science

The amount of male and female authors is approximated with the amount of author names that are believed to be male and female. However, the amount of names counted by gender is not exact.

This partly lays in how *dblp* registers data: First, identical names that belong to different authors are usually but not always marked with an integer number (dblp computer science bibliography 2019h). Second, an author might also use different aliases in different publications (dblp computer science bibliography 2019f). However there is no reason to believe that the way *dblp* registers data is biased towards men or women so while it could change the overall number of authors it does not influence the gender ratio.

The other part lays in the reliability of the classification results returned by *NamSor v2 API*. The high amount of names with a score lower than 5 leads to 1050103 of 2306418 not being classified. These unclassified authors could affect the overall result obtained with the analysis if the classifiable names are not a representative sample.

### 3 Analysis

First of all, low scores are produced by unisex, Asian and abbreviated names as well as pseudonyms. *NamSor* can not classify Asian names reliably, since many of those names are gender neutral or information on gender was lost when the name was transcribed into Latin letters (Mattauch et al. 2018, p. 4). Abbreviated names are given a low score and can even be classified incorrectly with a high score when the last name reveals the gender. “M. Myasnikova” for instance is mistakenly classified as male with a score of 9.06 while it should be classified as female since the name is Russian and has a female ending. Additionally it can be presumed that *NamSor API* is less certain when classifying a name as female as can be seen in Figure 2.6. The reason might lay in the data that *NamSor* uses to train its classification algorithms: For some parts of the world mostly data on men was available<sup>1</sup>. This means that women might have been classified as such, but in this thesis their classification was not considered to be reliable and thus they were not counted if classified with a low score.

All in all the sample of classified names is less international and some groups of authors might tend to abbreviate their first names more often. Plus, more female than male authors are not classified reliably, depending on the country of origin of a name. This could mean that the share of female authors is actually higher and the Gender Gap smaller.

## 3.2 Are groups of collaborating authors diverse, setting a good example for other areas of life?

### 3.2.1 Hypothesis

$h_0$  = Teams of collaborating authors consist of an equal amount of men and women.

$h_1$  = Teams of collaborating authors do not consist of an equal amount of men and women.

### 3.2.2 Condition

The hypothesis  $h_0$  is true if the median of differences between the amount of women and men per publication for publications with more than one author is not significantly different than expected.

---

<sup>1</sup> Personal correspondence with *NamSor*

#### 3.2.3 Method

The Sign Test can be used to test whether pairs of data are of significantly different value by comparing the median of all calculated differences  $D$  to 0 (Weerahandi 2003, p. 79). In the case of this hypothesis the Sign Test is used to test if the median of normed differences between the amount of women and men per publication is different than expected.

The following assumptions for the Sign Test have been extracted from (Weerahandi 2003, p. 79):

- Observations come in pairs, so their difference can be calculated
- Observations are continuously distributed

Those assumptions are met by the data available.

Of course, only publications written in a team should be tested. But before continuing with the test publications written in a team of authors where none of the authors could be classified reliably need to be removed from the testing data. For these teams the difference is always 0. If they were included in the testing data they would contribute to a higher amount of gender equal teams. After removing those invalid teams there are 3020710 publications that remain to be analyzed.

For each of those publications the normed difference between the amount of female and male authors needs is calculated. The differences need to be normed so they are independent from the team size. The normed difference  $d$  for a pair of data (amount of male contributors  $n_{male}$  and amount of female contributors  $n_{female}$ ) with an overall team size  $n$  is calculated with the following formula:

$$d = \frac{n(male)}{n} - \frac{n(female)}{n}$$

For example if there are 4 women in a team and 1 man, the normed difference is  $(\frac{n_{male}}{n} - \frac{n_{female}}{n} = 0.2 - 0.8 =) -0.6$  while the absolute difference is -3. If there are 8 women in a team and 2 men, the normed difference is still -0.6, but the absolute difference is -6.

Assumed that teams are mostly made up of an equal amount of men and women the expected median is 0.

The Sign Test is executed using the method `stats.wilcoxon` from *Python's SciPy* library. This test is originally meant to be used for the Wilcoxon Signed Rank Test that tests whether two

### 3 Analysis

samples are taken out of the same population. It can also be used for a simple Sign Test if fed with an array of differences  $D$  as a parameter. The method returns the test statistic  $W$  which is either the amount of differences greater than 0 or the amount of differences smaller than 0, depending on which of these numbers is smaller. The method additionally returns the two-sided p-value  $p$  (The Scipy community 2019). The p-value  $p$  is the probability for obtaining the returned Sign Test statistic  $W$  under the condition that the median of the testing data is 0. If  $p$  is smaller than the level of significance  $\alpha$ , the median of differences  $D$  is significantly different than 0.

Additionally, the histogram of the Gender Balance Indicator of publications with more than one author is presented<sup>2</sup>.

#### 3.2.4 Result

$$W(D, 0) = 74471634463.5$$

$$p = 0$$

$$\alpha = 0.05$$

The hypothesis  $h_0$  is rejected because  $p < \alpha$ . Teams of authors in the field of computer science are not composed diversely. Figure 3.2 shows that there are usually more men than women in a team. Teams of collaborating authors are by far not setting a good example for the way groups are made up in other parts of life, such as student life at university.

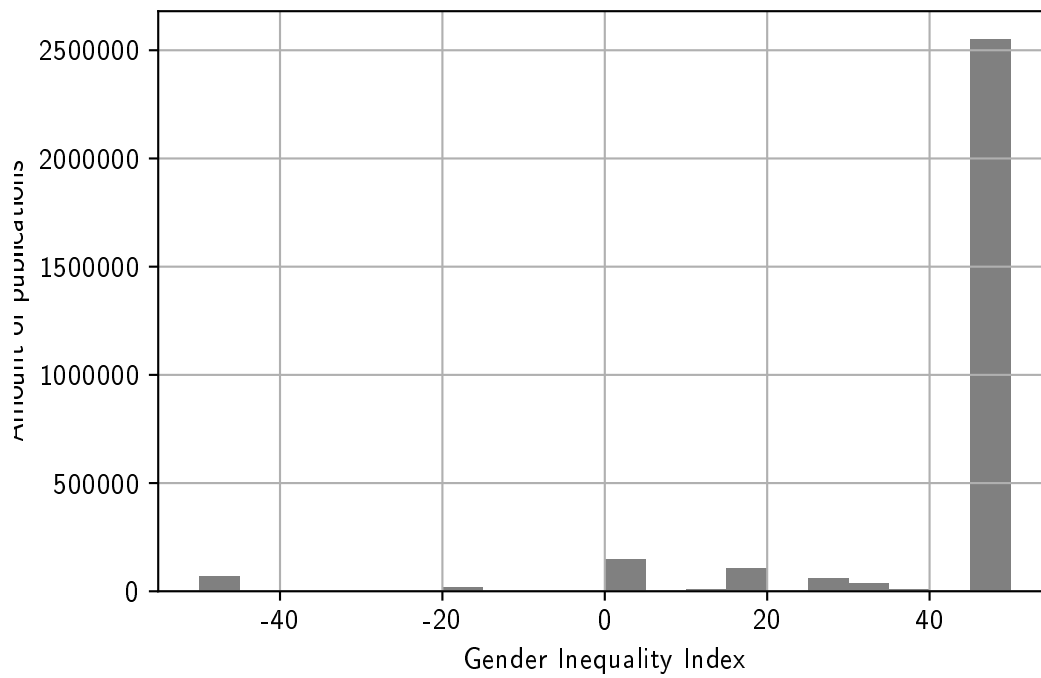
#### 3.2.5 Threads to validity: Approximation of the amount of men and women by team

The number of female and male authors per team used for analysis is not exact because the actual number depends on the gender of unclassified authors of the team. This threat to validity is elaborated further in the first hypothesis test.

---

<sup>2</sup> For an explanation of the Gender Balance Indicator see chapter 2 “Prerequisites”, section “Preparing data for analysis”

### 3 Analysis



**Figure 3.2:** Histogram of the Gender Balance Indicator of publications written in a team (excluding teams where no author could be classified by gender)

## 3.3 Is the amount of co-authors correlated to the diversity of a team?

### 3.3.1 Hypothesis

$h_0$  = Bigger teams are equally diverse as smaller teams.

$h_1$  = Bigger teams are more or less diverse than smaller teams.

### 3.3.2 Condition

The hypothesis  $h_0$  is true if the absolute Gender Balance Indicator does not correlate with the team size.



### 3 Analysis

#### 3.3.3 Method

Spearman's Rank Correlation Coefficient indicates whether and if so in which direction two sequences of ranked values correlate (Spearman 1904, p. 79–82). In this hypothesis it is used to calculate whether the absolute Gender Balance Indicator ([0,50]) correlates with the team size. The Gender Balance Indicator is taken for measurement as it is normed and thus is not directly influenced by the team size. Its absolute value is tested because the direction of bias is not of importance, only whether with a bigger team size the Gender Balance Indicator goes toward 0 or away from 0.

Only publications written by a team of authors where at least one author could be classified are taken into account. If no author was classified in a team these teams have a Gender Balance Indicator of 0 and could distort the result. After removing invalid teams 3020710 publications remain for analysis.

Spearman's Rank Correlation Coefficient  $\rho$  and the corresponding two-sided p-value  $p$  for the hypothesis that both sequences of data do not correlate is calculated using the method `stats.spearmanr` from *Python's SciPy* library for both sets of data. The method takes two arrays of data as parameters and automatically performs ranking and tie handling (The Scipy community 2018c).

A graph of the median per team size is drawn over a scatter graph of all Gender Balance Indicators using *Python's Matplotlib* library.

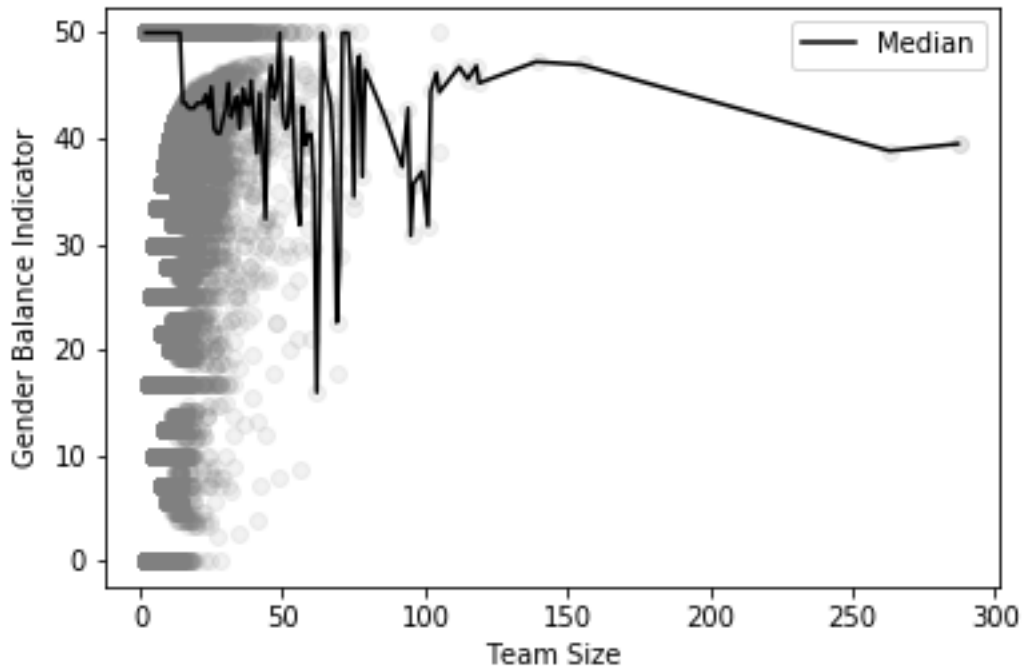
#### 3.3.4 Result

$$\rho(\text{Gender Balance Indicator, Team Size}) \approx -0.15$$

$$p = 0$$

$$\text{Level of significance } \alpha = 0.05$$

The correlation coefficient  $\rho$  indicates no correlation. However, the p-value  $p$  is smaller than  $\alpha$ , which indicates that there is no evidence to conclude that team size and Gender Balance Indicator are in fact not correlated. This means that it can not be stated that the size of a team of authors in the field of computer science does not correlate with the diversity of a team. Neither can it be said that there is correlation. Whether  $h_0$  is to be accepted or rejected can finally not be concluded. Which role the amount of co-authors plays for the diversity of a team remains open. Figure 3.3 suggests that the Gender Gap in Teams of authors is big for all sizes of teams.



**Figure 3.3:** Scatter graph to explore the question whether bigger teams are more diverse

#### 3.3.5 Threads to validity: Approximation of the amount of men and women by team

A constraint of validity for this test is the amount of unclassified authors. However, since this test only inquires about the relationship between team size and diversity, the amount of unclassified authors should not play into account per team size but maybe on a lower level when the Gender Balance Indicator is calculated. Better results may therefore be obtained with better data or another way than the Gender Balance Indicator of measuring gender inequality in a team.

### 3.4 Are female authors as productive as male authors?

#### 3.4.1 Hypothesis

$h_0$  = Female authors are as productive as male authors.

$h_1$  = Female authors are more or less productive than male authors.

### 3 Analysis

#### 3.4.2 Condition

The hypothesis  $h_0$  is true if the observed amounts of contributions are not significantly different for male and female authors.

#### 3.4.3 Method

The Mann Whitney U Test ranks and sorts data by size and calculates how often the data of one group is smaller than the data from the other group. When ranks are tied those are left out. The data does not need to be normally distributed (H. B. Mann 1947, p. 50f). Here the test is used to verify if the average amount of contributions made by male authors  $c_{male}$  and female authors  $c_{female}$  is significantly different. The available data meets all assumptions (See also (H. B. Mann 1947, p. 50f)):

- The measurement of the amount of contributions can be ranked.
- The observations and both compared groups (male and female authors) are independent.
- Data on the amount of contributions is continuous.

To obtain the contributions made by male and female authors,  $c_{male}$  and  $c_{female}$ , authors are approximated with the author names data set. All contributions per author name are taken into consideration if the name is classified with a score of 5 or higher.

The amount of contributions per author are calculated in the following way. Assuming one publication is worth one contribution. Then each author of a publication made  $\frac{1}{\text{Team Size}}$  contributions to it. For example, if 1 author worked in a team of size 5, their contribution has the size  $\frac{1}{5}$ . An author's total amount of contributions is the sum of all their contributions. Finally the amounts of contributions by author are grouped by gender, resulting in two groups of data  $c_{female}$  and  $c_{male}$ .

The method `stats.mannwhitneyu` from *Python's SciPy* Library is used to execute the Mann Whitney U Test. As parameters the method takes two arrays of data and a string indicating which p-value  $p$  to return (The Scipy community 2018b). Apart from  $p$  it also returns the  $U$  statistic.  $U$  indicates how often a value from the second data is smaller than a value from the first data set (H. B. Mann 1947, p. 50f). In this test, the data of contributions made by women  $c_{female}$  was entered as a first parameter and the contributions made by men  $c_{male}$  as a second one. Because of the order of the data,  $U$  signifies the amount of times a man's

### 3 Analysis

contribution is smaller than a woman's contribution. The "two-sided" p-value  $p$  was chosen to be returned. It signifies the probability of the resulting  $U$  statistic under the condition that  $h_0$  is true.

Mean, median and mode of both groups of data are calculated with the corresponding methods from *Python's Pandas* and *Statistics* libraries<sup>3</sup>.

#### 3.4.4 Result

$$U(c_{female}, c_{male}) = 69084463785.5$$

$$p = 0$$

$$\alpha = 0.05$$

Data	Mean	Median	Mode
$c_{male}$	2.24	0.54	0.33
$c_{female}$	1.38	0.5	0.33

**Table 3.1:** Mean, median and mode of the amount of contributions made by male and female authors

The hypothesis  $h_0$  is rejected because  $p$  is smaller than  $\alpha$ . Male and female authors are not equally productive when productivity means being on the list of authors, eventually together with a number of other scientists. From the descriptive statistics shown in Table 3.1 it appears that women are a little less productive than men in the field of computer science.

#### 3.4.5 Threads to validity: Approximation of the amount of productivity per author

The bigger the size of the team, the smaller the contribution of a single author. However, in reality it can not be expected that each author contributes the exact same amount of work to a publication. It is therefore not possible to say how big the contribution of an author to a publication really is. For an exact analysis of the amount of contributions, the amount of text written by an author might need to be measured or how much time they spent on research and report. However, this information is beyond the scope of the data registered by *dblp*. The constraints posed by the the amount of unclassifiable authors plays a role in this hypothesis as well and is further described in the first hypothesis test.

---

<sup>3</sup> <https://docs.python.org/3/library/statistics.html>

### 3.5 Did gender equality improve over time and can therefore an increase in the amount of role models be expected?

$h_0$  = Over time, the share of female authors stays the same.

$h_1$  = Over time, the share of female authors changes.

#### 3.5.1 Condition

The hypothesis  $h_0$  is correct if the percentage of female authors out of the total amount of authors by year does not correlate with the year.

#### 3.5.2 Method

Spearman's Rank Correlation Coefficient is to be used to calculate whether and if so by how much the share of female authors correlates with the year<sup>4</sup>.

In order for the percentage of female authors per year to be a reliable number, the amount of publications per year needs to be a big enough sample size. Since the total amount of publishing authors in the field of computer science per year is unknown, Cochran's formula can be used to determine the necessary sample size  $S$  (Cochran 1977, p. 75):

*Confidence level*  $c = 95\%$

*Z-Score for c*  $z = 2.58$

*Margin of Error*  $e = 0.05$

*Standard Deviation*  $d = 0.5$

$$S = \frac{z^2 \times d \times (1-d)}{e^2} = 665$$

Thus 665 publications define a sufficient sample size per year. Years before 1962 only have between 5 and 638 publications, so those are excluded. For 2019, only four months of data are available, so years after 2018 are to be excluded as well.

For the years 1962 to 2018, the share of contributing female authors per year of the total amount of authors contributing per year is calculated. The share is analyzed instead of the exact number of female authors because it can be expected that the amount of authors in

---

<sup>4</sup> Spearman's Rank Correlation Coefficient and the fitting *Python* method have been used in the third hypothesis test and are elaborated there

### 3 Analysis

total grows and therefore the amount of female authors grows as well - so only by looking at relative numbers a meaningful result can be retrieved.

The method `stats.spearmanr` from *Python's SciPy* library is used for the calculation of Spearman's Rank Correlation Coefficient  $\rho$  and the two-sided p-value  $p^5$ .  $p$  is the probability that the year  $y$  and the share of female authors  $s_{female}$  do not correlate.

Additionally, the timely development is visualized graphically.

#### 3.5.3 Result

$$\rho(y, s_{female}) = 0.98$$

$$p \approx 0$$

$$\alpha = 0.05$$

The hypothesis  $h_0$  is rejected because  $\rho$  shows a clear positive correlation between the year and the share of female authors. That  $p < \alpha$  supports the rejection of  $h_0$ . Figure 3.4 underlines this result. It can be assumed that over time, the share of female authors grows slowly. Figure 3.4 additionally shows that the share of contributions made by women is similarly correlating with the year. It can be estimated that the Gender Gap is closing slowly but steadily and that more female authors in the field of computer science are going to publish and be visible.

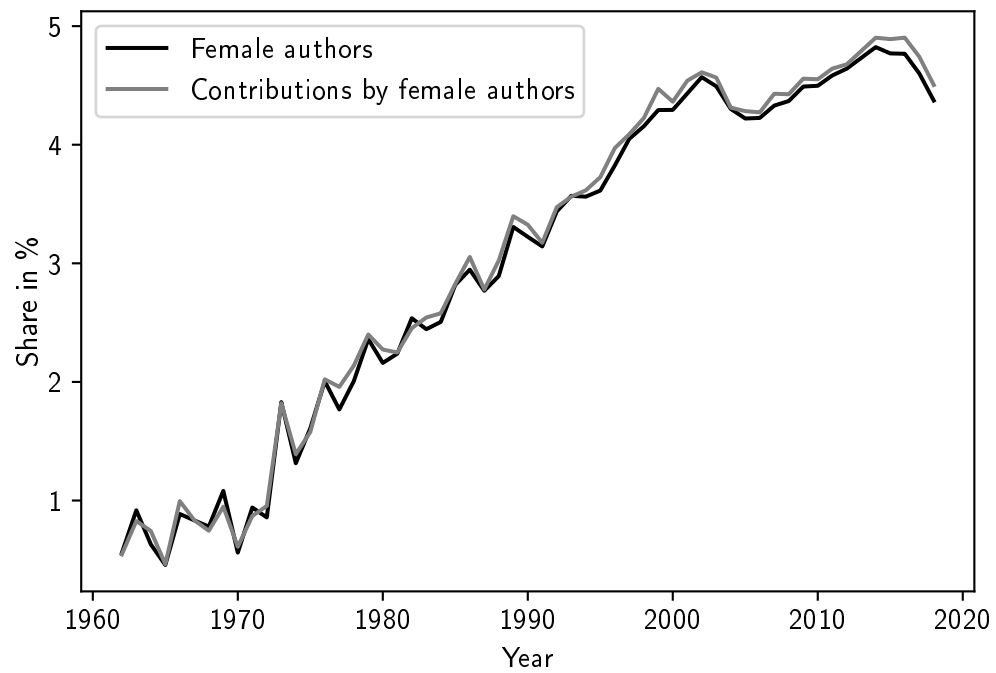
#### 3.5.4 Threads to validity: Approximation of the amount of male and female authors by year

Results concerning timely development are to be treated with caution because the amount of authors of unknown gender does grow over time ( $\rho = 0.83, p \approx 0$ ) as can be seen in Figure 3.5. This raise of unclassifiable names might be due to an increasing amount of Asian authors. So while a trend to grow can clearly be observed, the exact numbers are uncertain due to classification problems.

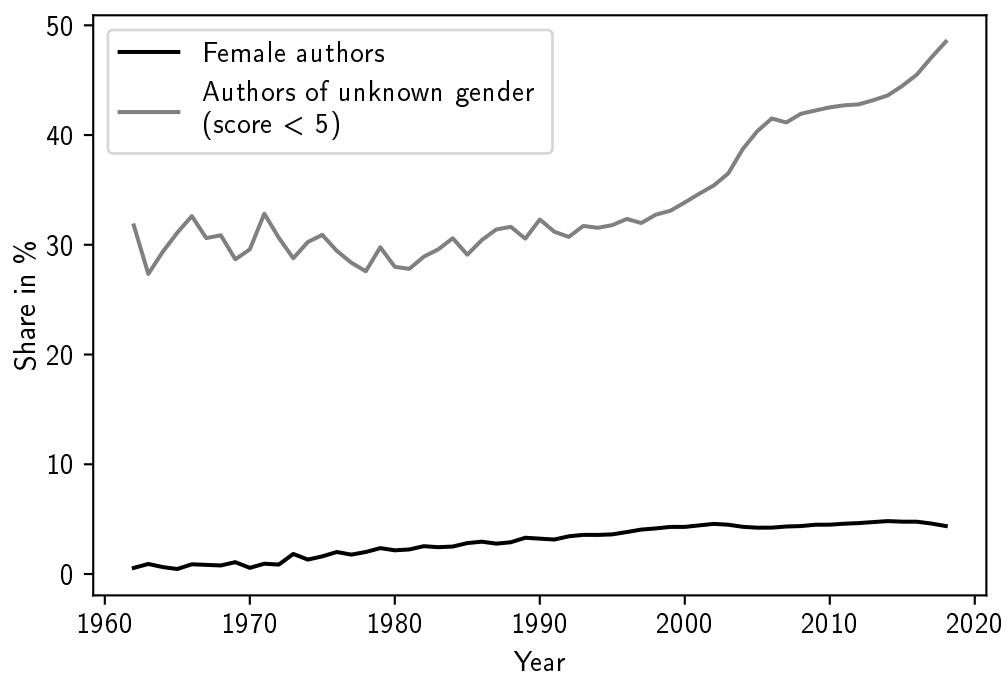
---

5 The use of this method is described in the third hypothesis test.

### 3 Analysis



**Figure 3.4:** Share of female authors and their contributions per year



**Figure 3.5:** Share of female and unclassified authors per year

## 4 Conclusion and future work

Research in the area of gender and computing is in the interest of all of today's and tomorrow's society and it is especially relevant for every computer scientist. Computer science is deeply webbed (literally) into humans' daily lives. A more diverse range of computer scientists could enable a broader range of people to experience the advantages of a digital, globalized world. Doing work in the field of gender and computing brings attention to the lack of diversity in computer science, which might encourage more people, including computer scientists, parents and politicians, to take action to close the Gender Gap, or any of the other gaps <sup>1</sup>. This thesis contributes to the mission of achieving Gender Equality in computer science, focusing on one part of the puzzle: computer scientists who publish. Because these are some of the people young scientists learn from they are important role models. The group of authors should therefore set an especially good example when it comes to diversity, just as teachers and mentors should. About publishing scientists this thesis found that...

- ... even though the Gender Gap in computer science authorship has been shrinking it is still too big
- ... teams of authors working together on a publication are not diverse
- ... women appear to be less productive authors in the field of computer science than their male counterparts if productivity is defined as being on the authors list of a publication together with a certain number of other authors

This thesis focused on answering questions concerning the state and development of gender equality in computer science publications with a data scientific approach and as such the detail of the available data places constraints on the accuracy and reliability of the results. Values are to be seen as an approximation: How many individual authors are behind the publications documented by *dblp*? What is really their gender? How reliable exactly is *NamSor*'s classification? How much did an author really contribute to a publication and thus how productive were they? All these are questions that go beyond the scope of the available

---

<sup>1</sup> More on diversity gaps in Computer Science in (Google Inc. 2016)



#### 4 Conclusion and future work

data and the scope of this thesis. This leaves the conclusion that what counts for the analysis of state and development of gender equality in computer science publications is the attention to general facts: Are there more male than female authors? Rather than: How many more male than female authors are there? Did the Gender Gap shrink? Rather than: By how much did the Gender Gap shrink? Are teams usually diverse? Rather than: How diverse are teams of authors?

The question whether or not team size plays a role for more diversity remains open despite its general formulation. In the future, this and many more questions may be answered using the data from *dblp* and *NamSor*, especially since the recently published version 2.0.5 of the API permits to better judge the reliability of a result. Some other questions could be: Is working in a team more common for male or for female authors? Are there areas of computer science where the state and development of gender equality in authorship are different? Going into the field could then provide researchers with the possibility of finding out reasons for any discovered correlations and reasons for gender equality where it was found. Using the *Jupyter Notebooks* created for this thesis the work done here could also be repeated in a few years to make note of any changes.

In the end, authors are only one part of the bigger picture. Lists of teachers in universities or schools could be analyzed for gender equality as well. Or officers in institutions of state that concern education or digital matters. Either way, the question that always needs to follow is: What can lead to the closing of the Gender Gap? And furthermore: What can I individually do to promote Gender Equality?

# Bibliography

Cochran, W. G. (1977). *Sampling Techniques (3th Edition)*, John Wiley & Sons.

**URL:** [https://archive.org/details/Cochran1977SamplingTechniques\\_201703/page/n87](https://archive.org/details/Cochran1977SamplingTechniques_201703/page/n87)

dblp computer science bibliography (2019a). How to parse dblp.xml? (Accessed on March 8 2019).

**URL:** <https://dblp.org/faq/1474681.html>

dblp computer science bibliography (2019b). Statistics - distribution of publication type. (Accessed on May 20 2019).

**URL:** <https://dblp.org/statistics/distributionofpublicationtype.html>

dblp computer science bibliography (2019c). Under what license is the data from dblp released? (Accessed on March 8 2019).

**URL:** <https://dblp.org/db/about/copyright.html>

dblp computer science bibliography (2019d). What do i find in dblp.xml? (Accessed on March 27 2019).

**URL:** <https://dblp.org/faq/16154937.html>

dblp computer science bibliography (2019e). What institution is behind dblp? (Accessed on March 8 2019).

**URL:** <https://dblp.org/faq/1474651.html>

dblp computer science bibliography (2019f). What is the meaning of these www-home-page records? (Accessed on May 20 2019).

**URL:** <https://dblp.org/faq/1474690.html>

dblp computer science bibliography (2019g). What is the scope of dblp? (Accessed on March 28 2019).

**URL:** <https://dblp.org/faq/1474671.html>

## Bibliography

- dblp computer science bibliography (2019h). Why are some names followed by a four digit number? (Accessed on May 20 2019).  
**URL:** <https://dblp.org/faq/1474704.html>
- Google (2014). Women who choose computer science—what really matters.  
**URL:** [https://docs.google.com/file/d/0B-E2rcvhnLQ\\_a1Q4VUxWQ2dtTHM/edit](https://docs.google.com/file/d/0B-E2rcvhnLQ_a1Q4VUxWQ2dtTHM/edit)
- Google Inc., G. I. (2016). Diversity gaps in computer science: Exploring the underrepresentation of girls, blacks and hispanics. (Accessed on January 28 2019).  
**URL:** <https://services.google.com/fh/files/misc/diversity-gaps-in-computer-science-report.pdf>
- H. B. Mann, D. R. W. (1947). On a test of whether one of two random variables is stochastically larger than the other, *The Annals of Mathematical Statistics* **18**(1): 50f.  
**URL:** <http://www.jstor.org/stable/2236101>
- Mattauch, S., Lohmann, K., Hannig, F., Lohmann, D. & Teich, J. (2018). The gender gap in computer science — a bibliometric analysis, *Technical Report CS-2018-02*, Friedrich-Alexander-Universität Erlangen-Nürnberg.
- NamSor (2018). Understanding namsor api precision for gender inference. (Accessed on March 27 2019).  
**URL:** <https://blog.namsor.com/2018/01/31/understanding-namsor-api-precision-for-gender-inference/>
- NamSor (2019a). Namsor api v2 documentation. (Accessed on March 28 2019).  
**URL:** <https://v2.namsor.com/NamSorAPIv2/apidoc.html>
- NamSor (2019b). Namsor api v2 python sdk license. (Accessed on March 27 2019).  
**URL:** <https://github.com/namsor/namsor-python-sdk2/blob/master/LICENSE>
- NamSor (2019c). openapi-client. (Accessed on July 20 2019).  
**URL:** <https://github.com/namsor/namsor-python-sdk2/blob/2440bca693bb8154fbbd9862e6657122db2d55ad/README.md>
- OECD (2016). Women in scientific production. (Accessed on July 16 2019).  
**URL:** <https://www.oecd.org/gender/data/women-in-scientific-production.htm>
- Open Knowledge Foundation (2018). Open data commons attribution license (odc-by) v1.0. (Accessed on March 8 2019).  
**URL:** <https://opendatacommons.org/licenses/by/1.0/>

## Bibliography

- Pearson, K. (1911). On the probability that two independent distributions of frequency are really samples from the same population, *Biometrika* 8(1/2): 250.  
**URL:** <http://www.jstor.org/stable/2331453>
- She Figures* (2015). *Technical report*. (Accessed on July 8 2019).  
**URL:** [https://ec.europa.eu/research/swafs/pdf/pub\\_gender\\_equality/she\\_figures\\_2015-final.pdf](https://ec.europa.eu/research/swafs/pdf/pub_gender_equality/she_figures_2015-final.pdf)
- Spearman, C. (1904). The proof and measurement of association between two things, *The American Journal of Psychology* 15(1): 79–82.  
**URL:** <http://www.jstor.org/stable/1412159>
- The Scipy community (2018a). Scipy v1.1.0 reference guide: scipy.stats.chisquare. (Accessed on July 15 2019).  
**URL:** <https://docs.scipy.org/doc/scipy-1.1.0/reference/generated/scipy.stats.chisquare.html>
- The Scipy community (2018b). Scipy v1.1.0 reference guide: scipy.stats.mannwhitneyu. (Accessed on July 15 2019).  
**URL:** <https://docs.scipy.org/doc/scipy-1.1.0/reference/generated/scipy.stats.mannwhitneyu.html>
- The Scipy community (2018c). Scipy v1.1.0 reference guide: scipy.stats.spearmanr. (Accessed on July 15 2019).  
**URL:** <https://docs.scipy.org/doc/scipy-1.1.0/reference/generated/scipy.stats.spearmanr.html>
- The Scipy community (2019). Scipy v1.3.0 reference guide: scipy.stats.wilcoxon. (Accessed on July 15 2019).  
**URL:** <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.wilcoxon.html>
- Weerahandi, S. (2003). *Exact Statistical Methods for Data Analysis*, Springer Series in Statistics, Springer New York.  
**URL:** <https://books.google.de/books?id=F7ISekNhKZ0C>

# Statutory Declaration

"I herewith formally declare that I have written the submitted thesis independently. I did not use any outside support except for the quoted literature and other sources mentioned in the paper. I clearly marked and separately listed all of the literature and all of the other sources which I employed when producing this academic work, either literally or in content. I am aware that the violation of this regulation will lead to failure of the thesis."

Linda Fernsel, s0555949

Berlin, July 24, 2019      Signature \_\_\_\_\_