# R Notebook

## Intro

This corpus assembles the texts published in the DIGAR (Digital Archive) database of the Estonian National Library, that can be freely shared.

### Corpus processing

Corpus comprises of print publications that were represented and accessible via the Estonian National Bibliography. Raw .txt files simply contain the text exported from the pdf-s (via pdf2txt). Unless the text was “born digital”, these texts contain the automatically transcribed texts via OCR over a long period of time. The quality of the text depends on the quality of the processed image and the tools available when the text was recognized. While these texts are not suitable for all types of study, there are many possibilities for used even with 90%, 75% or 50% accuracy. The accuracy may also be improved in future editions.

The processed texts have been tokenized and analyzed with the EstNLTK python package (v. 1.4.1, [**?**]ORASMAA16.332). This has been done with two regimes: 1) ‘simple’ - with guessing turned off - to distinguish definitely recognizable wordforms from non-standard spellings and bad OCR, 2) ‘guess’ - with guessing turned on - to give a best guess for the identity of the word based on the contextual clues used in EstNLTK. Before tokenization, special characters (except for ”„“_ -.,;”:?()õäöüÕÄÖÜ[]/) were removed from the texts, and characters with diacritics that do not usually occur in Estonian language texts and may be OCR artefacts were simplified (e.g. î -> i, è -> e, ç -> c). Additionally, because OCR texts often mistakenly inserted blanks within words (e.g. s õ r e n d a t u d) in some lines, in this case using two or more blanks for word separation, the presence of multiple 1-letter words or multiple 2-blank separations was checked on each line. If this was the case, the shorter blanks were removed and longer blanks relied on for word boundaries. This may have in some cases joined some words, and was unable to find all the examples, however improved the processing of the corpus.

The processed texts are stored as .tsv-s that contain the variables posted below (categories described here - https://estnltk.github.io/estnltk/1.4.1/tutorials/text.html#morphological-analysis). The suffix of the variable indicates the regime used.

- word_texts_simple
- lemmas_simple
- postag_descriptions_simple
- word_texts_guess
- lemmas_guess
- roots_guess
- root_tokens_guess
- forms_guess
- endings_guess
- postags_guess
- postag_descriptions_guess

Whether this text processing helps your analysis depends on your particular question.

### Overview of the processed corpus

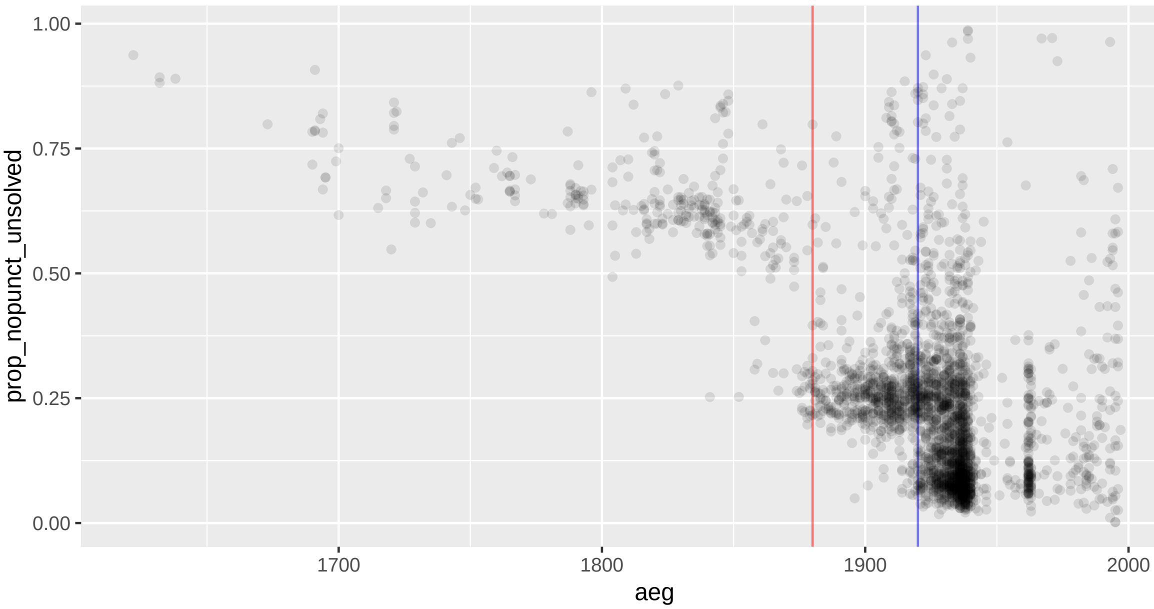
```
## -- Attaching packages --------------------------------------------------------- tidyverse 1.2.1 --
## √ ggplot2 3.0.0     √ purrr   0.2.5
## √ tibble  1.4.2     √ dplyr   0.7.6
## √ tidyr   0.8.1     √ stringr 1.3.1
## √ readr   1.1.1     √ forcats 0.3.0

## -- Conflicts ------------------------------------------------------ tidyverse_conflicts() --
## x dplyr::between()   masks data.table::between()
## x dplyr::filter()    masks stats::filter()
## x dplyr::first()     masks data.table::first()
## x dplyr::lag()       masks stats::lag()
## x dplyr::last()      masks data.table::last()
## x purrr::transpose() masks data.table::transpose()
```

## OCR and lemmatization success

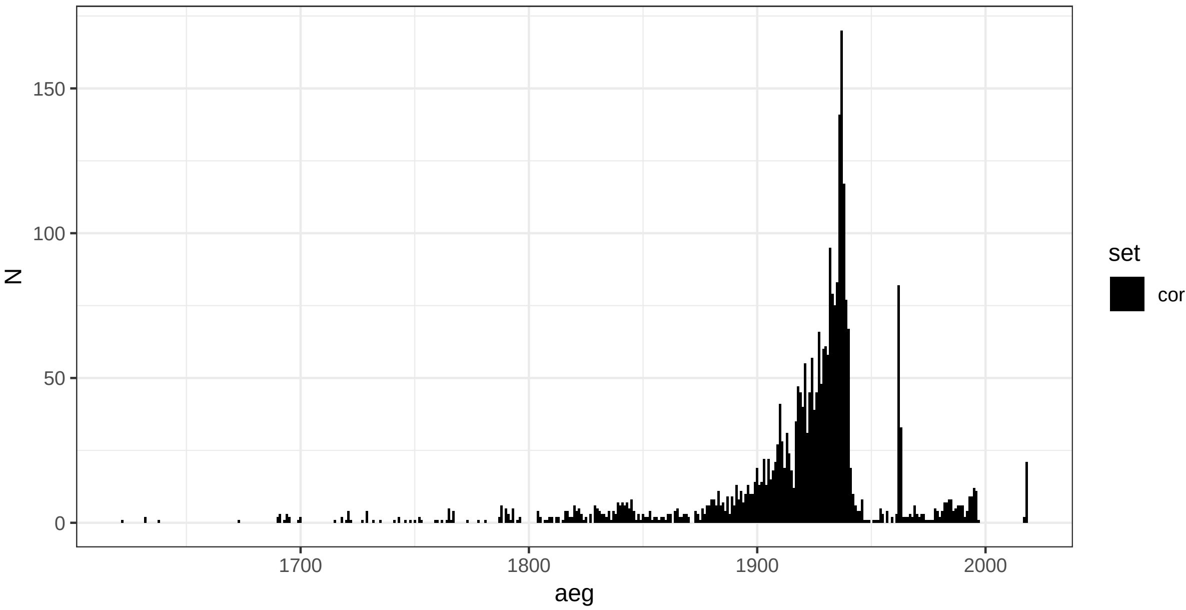
Two periods:

1) Before 1880s, "old writing tradition".
2) Before 1920s, w commonly used instead of V.

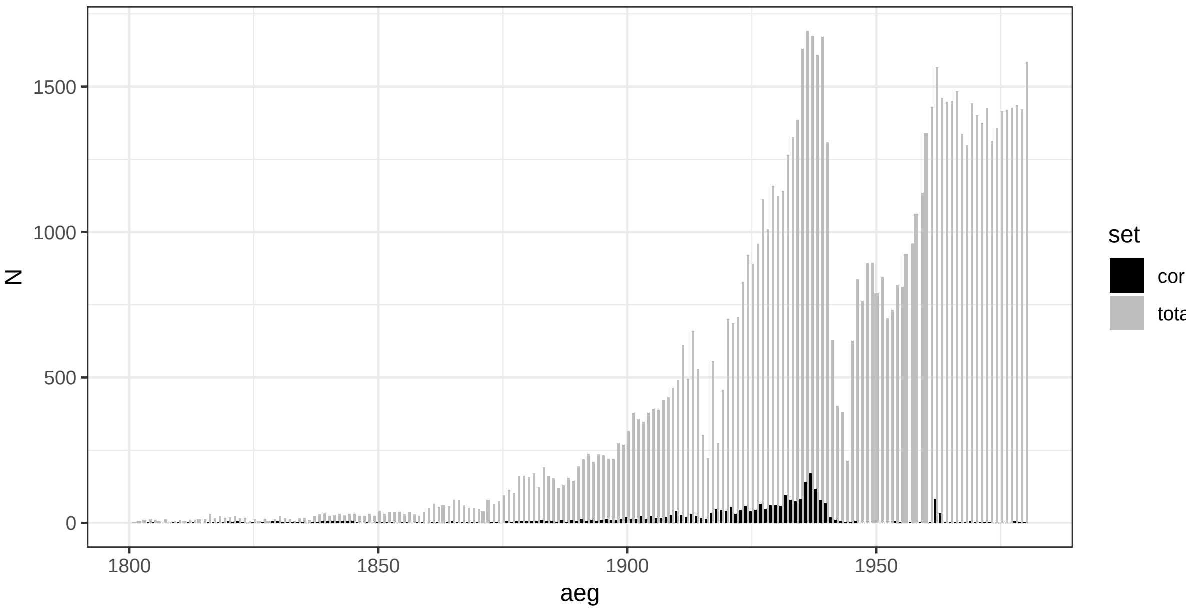


annotation success-1.bb

The distribution of texts over time, and compared to the registered prints.

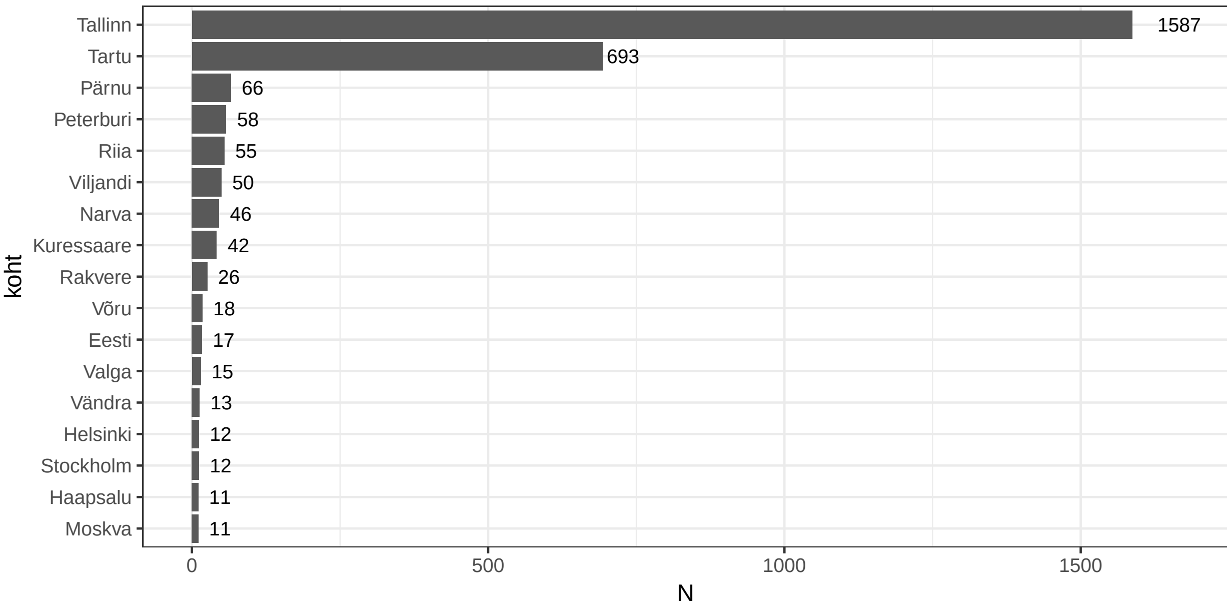


corpus overview-1.bb

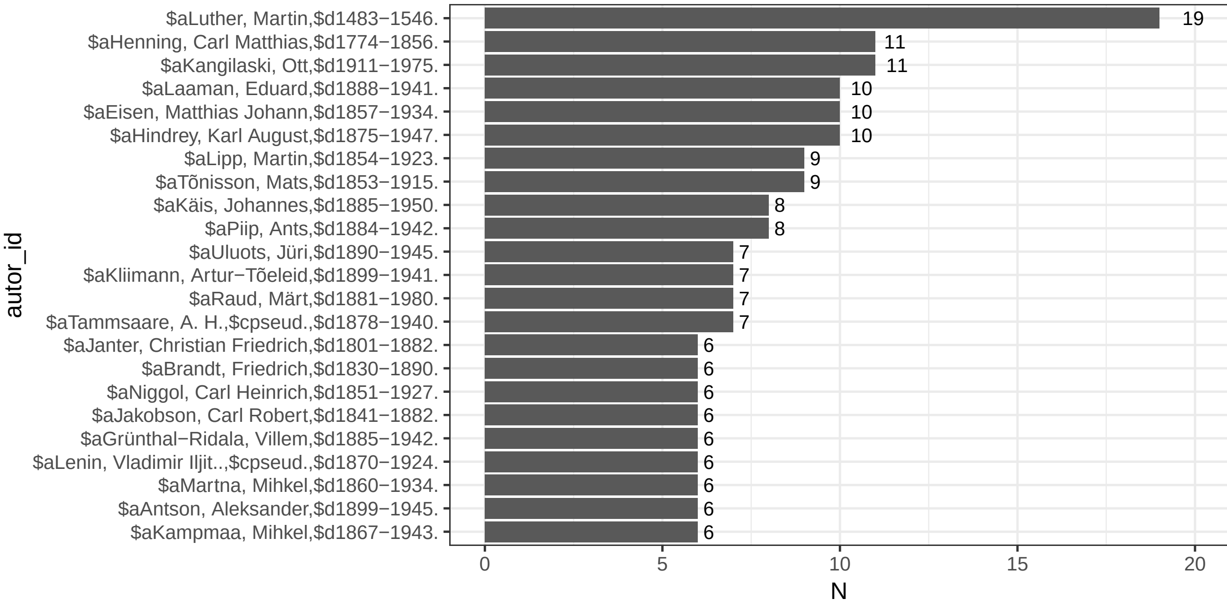


corpus overview-2.bb

The distribution of books between cities where they were published.

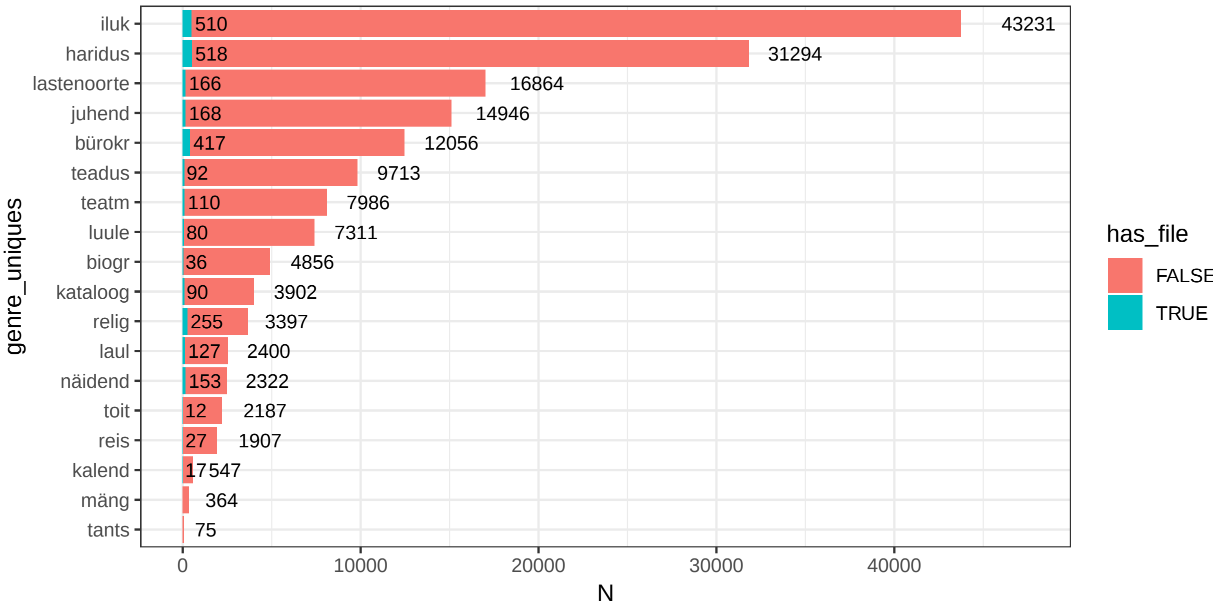


corpus cities-1.bb



corpus cities-2.bb

The distribution of texts across genres, and as proportion of registered texts.



corpus genres-1.bb

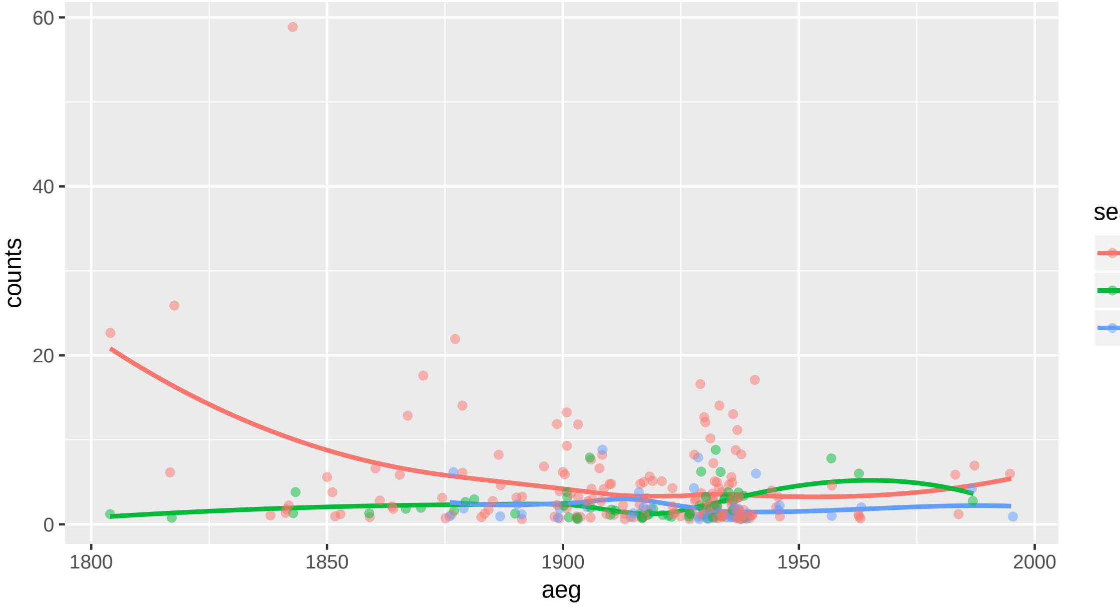
## Usage

A possible rule of thumb can be to take only texts with at least 50 recognized tokens within text.

How much unsuccessful OCR can interfere with your processing depends on your goals. For tracking the presence of particular keywords, the OCR texts can work quite well. For tracking more general measures, like type-token ratio, unsuccessful OCR can cause significant problems.

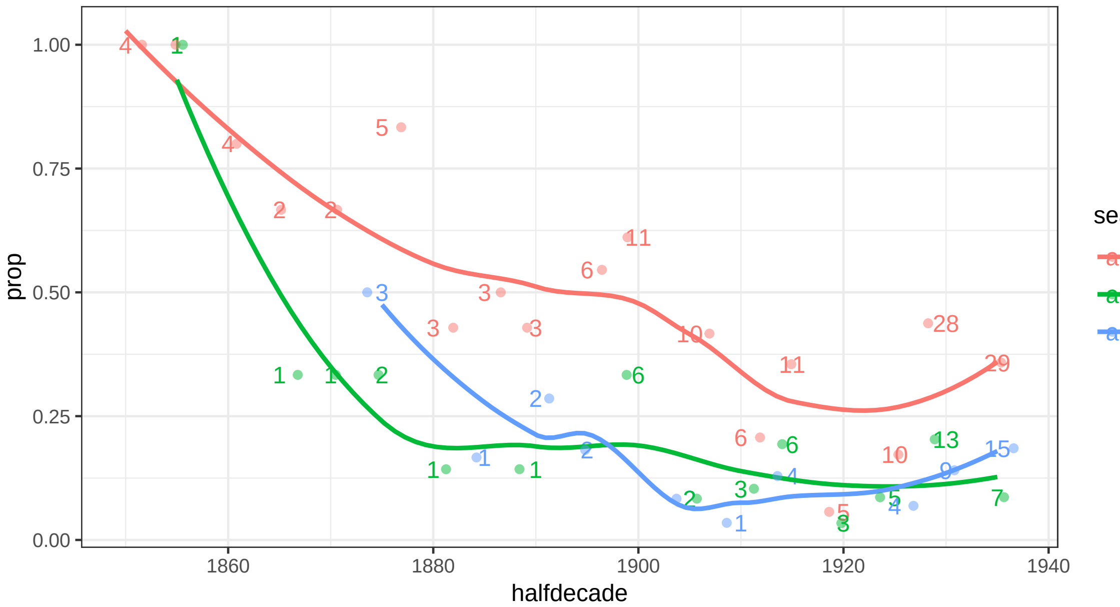
Included in this is an example of how to search for particular keywords within the text, and plot the findings. Both simple and averages per year. Or number of texts within the corpus that year, or number of tokens within the corpus that year...

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



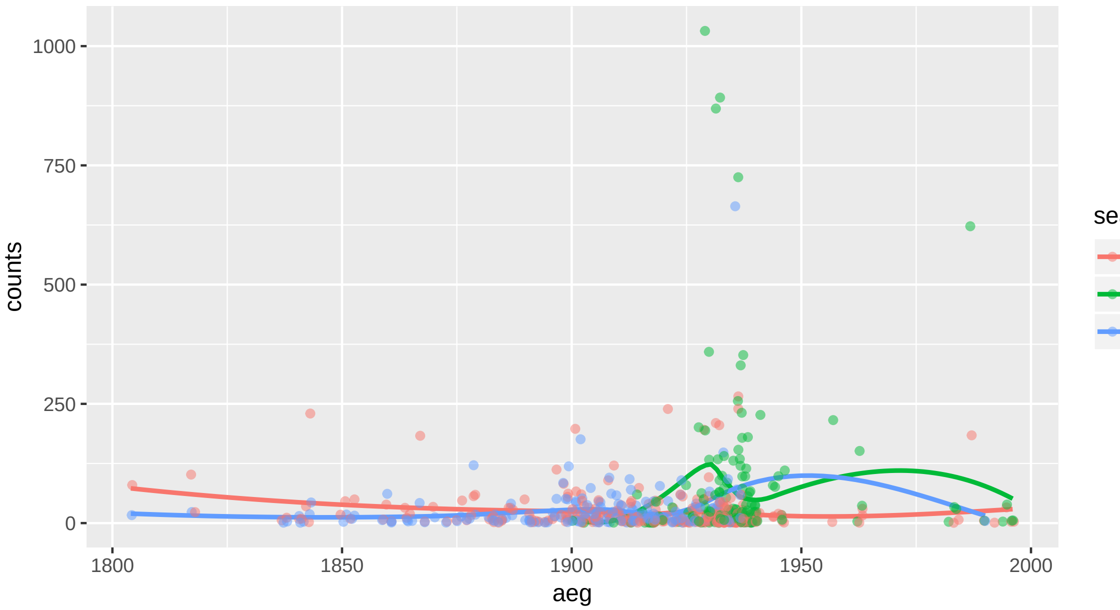
example use case too?-1.bb

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



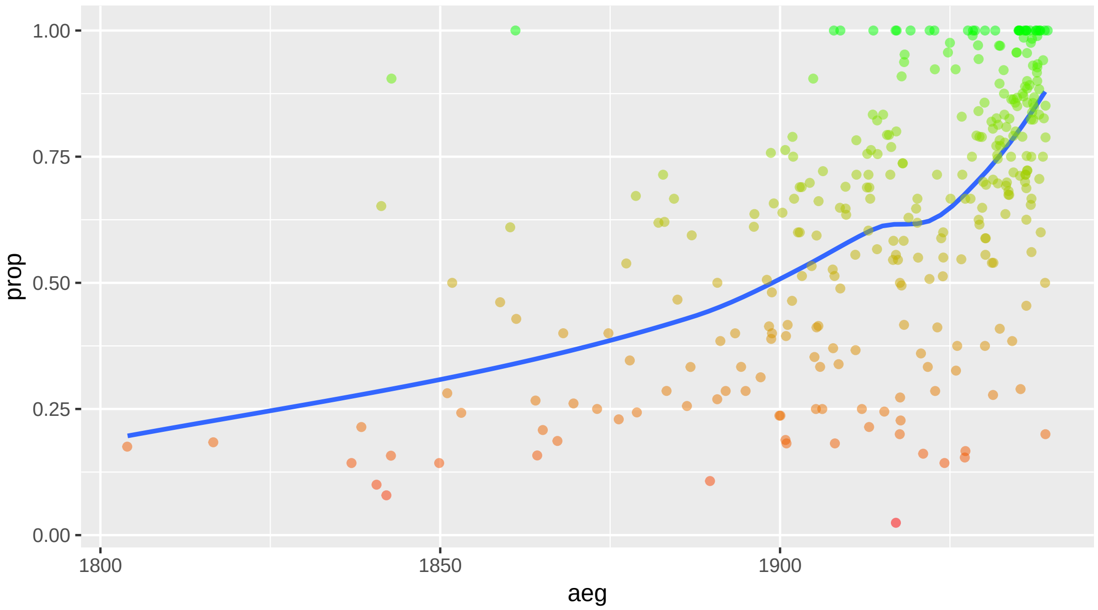
example use case too?-2.bb

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



example use case too?-3.bb

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



example use case too?-4.bb

@InProceedings{ORASMAA16.332, author = {Siim Orasmaa and Timo Petmanson and Alexander Tkachenko and Sven Laur and Heiki-Jaan Kaalep}, title = {EstNLTK - NLP Toolkit for Estonian}, booktitle = {Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)}, year = {2016}, month = {may}, date = {23-28}, location = {Portorož, Slovenia}, editor = {Nicoletta Calzolari (Conference Chair) and Khalid Choukri and Thierry Declerck and Marko Grobelnik and Bente Maegaard and Joseph Mariani and Asuncion Moreno and Jan Odijk and Stelios Piperidis}, publisher = {European Language Resources Association (ELRA)}, address = {Paris, France}, isbn = {978-2-9517408-9-1}, language = {english} }