

# R Notebook, Python und UTF-8

Jörg Kantel

2016-10-20

## Python im R Notebook – ein erster Test

Bei meinen Experimenten mit dem neuen R Notebook habe ich nicht nur dieses, sondern auch das Tufte-Paket<sup>1</sup> für einen schöneren HTML- und PDF-/LaTeX-Output entdeckt. Und so habe ich beides zusammen für meine ersten Experimente mit dem R Notebook und Python genutzt.

Leider verbindet sich das Notebook zur Zeit nur mit dem Python, was R unter `Sys.which("python")` findet<sup>2</sup>, und das ist leider das (System-) Python 2.7.5 von Apple, nicht jedoch mein Anaconda Python 3.5. Wie dem auch sei, für erste Tests muß das reichen.

```
# -*- coding: utf-8 -*-
from collections import Counter
path = "data/aufklaerung.txt"

with open(path , mode="r") as f:
    text = f.read()
print(text)

text= text.lower()
cntr = Counter(text.split())
liste = cntr.most_common()
print(liste[0:5])
```

```
## Beantwortung der Frage: Was ist Aufklärung?
## Aufklärung ist der Ausgang des Menschen aus seiner selbst verschuldeten Unmündigkeit.
## [('der', 2), ('ist', 2), ('des', 1), ('unm\xc3\xbcndigkeit.', 1), ('frage:', 1)]
```

So schön das Tufte-Paket auch ist, man sieht auch gleich seine erste Unzulänglichkeit: Breite Codezeilen sind zumindest beim HTML-Output nur mit dem ungeliebten, horizontalen Scrollbalken möglich (beim PDF-Output sieht die Geschichte schon besser aus).

Nimmt man dagegen als Output das »normale« HTML-Dokument des Notebooks, ist eigentlich alles viel schicker. Wenn man keinen guten Grund hat, sollte man sich daher wirklich überlegen, ob man die Funktionalität der Ästhetik opfert.

Wie gesagt, im Druck sieht alles schicker aus. Der *Tufte-Handout-Style* ist eben für Druckerzeugnisse entworfen und nicht für Online-Publikationen. Das merkt man deutlich.

<sup>1</sup> <http://rstudio.github.io/tufte/>

<sup>2</sup> Das scheint eine Eigenschaft der Knitr-Language-Engine zu sein

