

Programming theologians

Text-Fabric (<https://github.com/ETCBC/text-fabric>): Ancient texts as fabrics of source and annotations.

data model (<https://github.com/ETCBC/text-fabric/wiki/Data-model>): Text objects, relationships, features.

Got it? Get it!

home page (<https://github.com/ETCBC/text-fabric/wiki>)



Join the computing gang

1. go to <https://shebanq.jove.surfsara.nl> (<https://shebanq.jove.surfsara.nl>) and log in (see paper ticket)
2. select assignment prog_theo, fetch leipzig and click it
3. click `Programming theologians.ipynb` and off-you-go

The screenshot shows the JupyterLab interface. At the top left is the Jupyter logo. On the top right are 'Control Panel' and 'Logout' buttons. Below these are tabs for 'Files', 'Running', 'Clusters', and 'Assignments'. The 'Assignments' tab is active. It shows a header 'Released, downloaded, and submitted assignments for course: prog_theo' with a dropdown menu and a refresh icon. The dropdown menu is open, showing 'counting', 'nbcourse', and 'prog_theo'. Below this is a table with three sections: 'Released assignments' containing 'leipzig', 'Downloaded assignments' with the message 'There are no downloaded assignments.', and 'Submitted assignments' with the message 'There are no submitted assignments.'. A blue 'Fetch' button is located to the right of the 'Released assignments' section.

jupyter

Control Panel Logout

Files Running Clusters Assignments

Released, downloaded, and submitted assignments for course: prog_theo

Released assignments

leipzig

Downloaded assignments

There are no downloaded assignments.

Submitted assignments

There are no submitted assignments.

Fetch

Before the beginning

In [1]:

```
import collections

from IPython.display import display

import matplotlib.pyplot as plt
%matplotlib inline

import pandas
pandas.set_option('display.notebook_repr_html', True)
```

API to the Hebrew Text

In [2]:

```
from tf.fabric import Fabric

ETCBC = 'hebrew/etcbc4c'
PHONO = 'hebrew/phono'

TF_H = Fabric( modules=[ETCBC, PHONO], silent=False )
```

This is Text-Fabric 2.3.7
Api reference : <https://github.com/ETCBC/text-fabric/wiki/Api>
(<https://github.com/ETCBC/text-fabric/wiki/Api>)
Tutorial : <https://github.com/ETCBC/text-fabric/blob/master/docs/tutorial.ipynb> (<https://github.com/ETCBC/text-fabric/blob/master/docs/tutorial.ipynb>)
Data sources : <https://github.com/ETCBC/text-fabric-data>
(<https://github.com/ETCBC/text-fabric-data>)
Data docs : <https://etcbc.github.io/text-fabric-data>
(<https://etcbc.github.io/text-fabric-data>)
Shebanq docs : <https://shebanq.ancient-data.org/text>
(<https://shebanq.ancient-data.org/text>)
Slack team : <https://shebanq.slack.com/signup>
(<https://shebanq.slack.com/signup>)
Questions? Ask shebanq@ancient-data.org for an invite to Slack
111 features found and 0 ignored

In [3]:

```
apiH = TF_H.load('sp')

0.00s loading features ...
| 0.17s B sp from /Users/dirk/github/text-fab
ric-data/hebrew/etcbc4c
| 0.00s Feature overview: 104 for nodes; 5 for edges; 2 configs
; 7 computed
5.76s All features loaded/computed - for details use loadLog()
```

API to the Greek Text

In [4]:

```
TF_G = Fabric(modules='greek/sblgnt')
```

This is Text-Fabric 2.3.7

Api reference : <https://github.com/ETCBC/text-fabric/wiki/Api>

(<https://github.com/ETCBC/text-fabric/wiki/Api>)

Tutorial : [https://github.com/ETCBC/text-fabric/blob/master/docs/](https://github.com/ETCBC/text-fabric/blob/master/docs/tutorial.ipynb)

[tutorial.ipynb](https://github.com/ETCBC/text-fabric/blob/master/docs/tutorial.ipynb) ([https://github.com/ETCBC/text-fabric/blob/master/docs/](https://github.com/ETCBC/text-fabric/blob/master/docs/tutorial.ipynb)

[tutorial.ipynb](https://github.com/ETCBC/text-fabric/blob/master/docs/tutorial.ipynb))

Data sources : <https://github.com/ETCBC/text-fabric-data>

(<https://github.com/ETCBC/text-fabric-data>)

Data docs : <https://etcbc.github.io/text-fabric-data>

(<https://etcbc.github.io/text-fabric-data>)

Shebanq docs : <https://shebanq.ancient-data.org/text>

(<https://shebanq.ancient-data.org/text>)

Slack team : <https://shebanq.slack.com/signup>

(<https://shebanq.slack.com/signup>)

Questions? Ask shebanq@ancient-data.org for an invite to Slack

63 features found and 0 ignored

In [5]:

```
apiG = TF_G.load('psp')
```

```
0.00s loading features ...
```

```
| 0.05s B psp from /Users/dirk/github/text-fab
```

```
ric-data/greek/sblgnt
```

```
| 0.00s Feature overview: 60 for nodes; 2 for edges; 1 configs;
```

```
7 computed
```

```
1.73s All features loaded/computed - for details use loadLog()
```

Easy switching

In [6]:

```
def doGreek():
    global T
    global L
    global F
    global Fs
    T = apiG.T
    L = apiG.L
    F = apiG.F
    Fs = apiG.Fs

def doHebrew():
    global T
    global L
    global F
    global Fs
    T = apiH.T
    L = apiH.L
    F = apiH.F
    Fs = apiH.Fs

def doingHebrew(): return F is apiH.F

def doingGreek(): return F is apiG.F
```

In the beginning

The first verse

In Hebrew

In [7]:

```
doHebrew()
```

In [8]:

```
T.text(range(1,12))
```

Out[8]:

' בְּרֵאשִׁית בָּרָא אֱלֹהִים אֶת הַשָּׁמַיִם וְאֶת הָאָרֶץ: '

In [9]:

```
T.text(range(1,12), fmt='text-phono-full')
```

Out[9]:

' b^ərēš ,îṭ bār 'ā ?^əlōh 'îm ? ,ēṭ haššām ,ayim w^ə? ,ēṭ hā? 'āreš . '

In [10]:

```
T.formats
```

Out[10]:

```
{'lex-orig-full',  
 'lex-orig-plain',  
 'lex-trans-full',  
 'lex-trans-plain',  
 'text-orig-full',  
 'text-orig-full-ketiv',  
 'text-orig-plain',  
 'text-phono-full',  
 'text-trans-full',  
 'text-trans-full-ketiv',  
 'text-trans-plain'}
```

In [11]:

```
T.text(range(1,12), fmt='lex-orig-plain')
```

Out[11]:

```
' ב ראשית ברא אלהים את ה שמימ ו את ה ארצ '
```

In Greek

In [12]:

```
doGreek()
```

In [13]:

```
firstVerse = T.nodeFromSection(('Matthew', 1, 1))  
F.otype.v(firstVerse)
```

Out[13]:

```
'verse'
```

In [14]:

```
words = L.d(firstVerse, otype='word')  
words
```

Out[14]:

```
[1, 2, 3, 4, 5, 6, 7, 8]
```

In [15]:

```
T.text(words)
```

Out[15]:

```
'Βίβλος γενέσεως Ἰησοῦ χριστοῦ υἱοῦ Δαυὶδ υἱοῦ Ἀβραάμ. '
```

In [16]:

```
T.formats
```

Out[16]:

```
{'lex-orig-full', 'text-orig-full', 'text-orig-plain'}
```

In [17]:

```
T.text(words, fmt='text-orig-plain')
```

Out[17]:

```
'ΒΙΒΛΟΣ ΓΕΝΕΣΕΩΣ ΙΗΣΟΥ ΧΡΙΣΤΟΥ ΥΙΟΥ ΔΑΥΙΔ ΥΙΟΥ ΑΒΡΑΑΜ. '
```

In [18]:

```
T.text(words, fmt='lex-orig-full')
```

Out[18]:

```
'βίβλος γένεσις Ἰησοῦς Χριστός υἱός Δαυίδ υἱός Ἀβραάμ '
```

Man and woman

God created the genders, we count them.

Which genders have we?

In [19]:

```
doHebrew()
```

```
TF_H.load('gn', add=True)
```

```
0.00s loading features ...
|      0.21s B gn          from /Users/dirk/github/text-fab
ric-data/hebrew/etc4c
0.22s All additional features loaded - for details use loadLog()
```

In [20]:

```
{F.gn.v(w) for w in F.otype.s('word')}
```

Out[20]:

```
{'NA', 'f', 'm', 'unknown'}
```

Genders in Hebrew and Greek

In [21]:

```
doGreek()  
TF_G.load('Gender', add=True)  
  
0.00s loading features ...  
|      0.06s B Gender          from /Users/dirk/github/text-fab  
ric-data/greek/sblgnt  
0.07s All additional features loaded - for details use loadLog()
```

In [22]:

```
def getGenders():  
    featureName = 'gn' if doingHebrew() else 'Gender'  
    return {Fs(featureName).v(w) for w in F.otype.s('word')}
```

In [23]:

```
doGreek()  
print('Greek: {}'.format(getGenders()))  
doHebrew()  
print('Hebrew: {}'.format(getGenders()))
```

```
Greek: {None, 'Neuter', 'Feminine', 'Masculine'}  
Hebrew: {'NA', 'm', 'f', 'unknown'}
```

Counting genders

In [24]:

```
def countGenders():  
    featureName = 'gn' if doingHebrew() else 'Gender'  
    stats = collections.Counter()  
    for w in F.otype.s('word'):  
        stats[Fs(featureName).v(w)] += 1  
    print(stats)  
countGenders()
```

```
Counter({'NA': 180152, 'm': 164191, 'unknown': 45524, 'f': 36714})
```

... in graphic detail ...

In [25]:

```
def genderBias(book):
    bookNode = T.nodeFromSection((book,))
    chapterNodes = L.d(bookNode, otype='chapter')
    x = [T.sectionFromNode(c)[1] for c in chapterNodes]
    masc = dict((c, 0) for c in x)
    fem = dict((c, 0) for c in x)
    neut = dict((c, 0) for c in x)
    absent = dict((c, 0) for c in x)
    total = dict((c, 0) for c in x)

    genderFeature = 'gn' if doingHebrew() else 'Gender'

    for chapterNode in chapterNodes:
        chapter = T.sectionFromNode(chapterNode)[1]
        words = L.d(chapterNode, otype='word')
        for w in words:
            total[chapter] += 1
            gender = Fs(genderFeature).v(w)
            if gender in {'m', 'Masculine'}: masc[chapter] += 1
            if gender in {'f', 'Feminine'}: fem[chapter] += 1
            if gender in {'Neuter'}: neut[chapter] += 1
            if gender in {'NA', 'unknown', None}: absent[chapter] += 1
    m = [100 * masc[c] / total[c] for c in x]
    f = [100 * fem[c] / total[c] for c in x]
    n = [100 * neut[c] / total[c] for c in x]
    a = [100 * absent[c] / total[c] for c in x]

    fig = plt.figure()
    plt.plot(x, m, 'b-', x, f, 'r-', x, n, 'g-', x, a, '0.5')
    plt.axis([x[0], x[-1], 0, 70])
    plt.xticks(x, x, rotation='vertical')
    plt.margins(0.2)
    plt.subplots_adjust(bottom=0.15);
    plt.title('gender in {} {}-{}'.format(book, x[0], x[-1]))
```

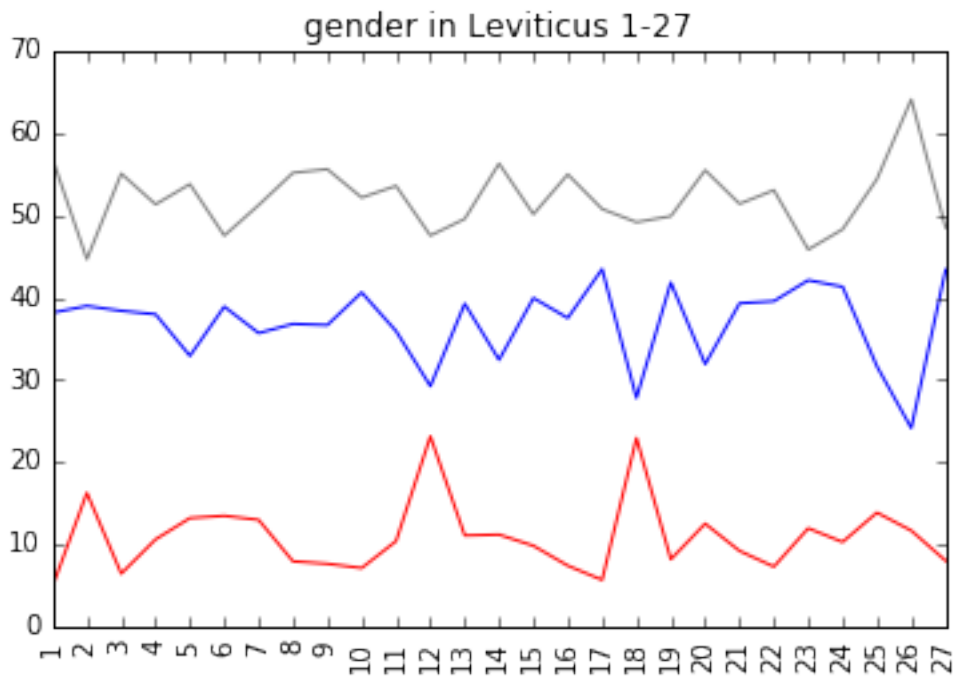
In [26]:

```
print(', '.join(T.sectionFromNode(b)[0] for b in F.otype.s('book')))
```

Genesis, Exodus, Leviticus, Numbers, Deuteronomy, Joshua, Judges, 1_Sa
muel, 2_Samuel, 1_Kings, 2_Kings, Isaiah, Jeremiah, Ezekiel, Hosea, Jo
el, Amos, Obadiah, Jonah, Micah, Nahum, Habakkuk, Zephaniah, Haggai, Z
echariah, Malachi, Psalms, Job, Proverbs, Ruth, Song_of_songs, Ecclesi
astes, Lamentations, Esther, Daniel, Ezra, Nehemiah, 1_Chronicles, 2_C
hronicles

In [27]:

```
genderBias('Leviticus')
```



Inspect some peaks and dips

In [28]:

```
TF_H.load('gloss', add=True)
```

```
0.00s loading features ...
|      0.04s B gloss          from /Users/dirk/github/text-fab
ric-data/hebrew/etc4c4c
0.04s All additional features loaded - for details use loadLog()
```

In [29]:

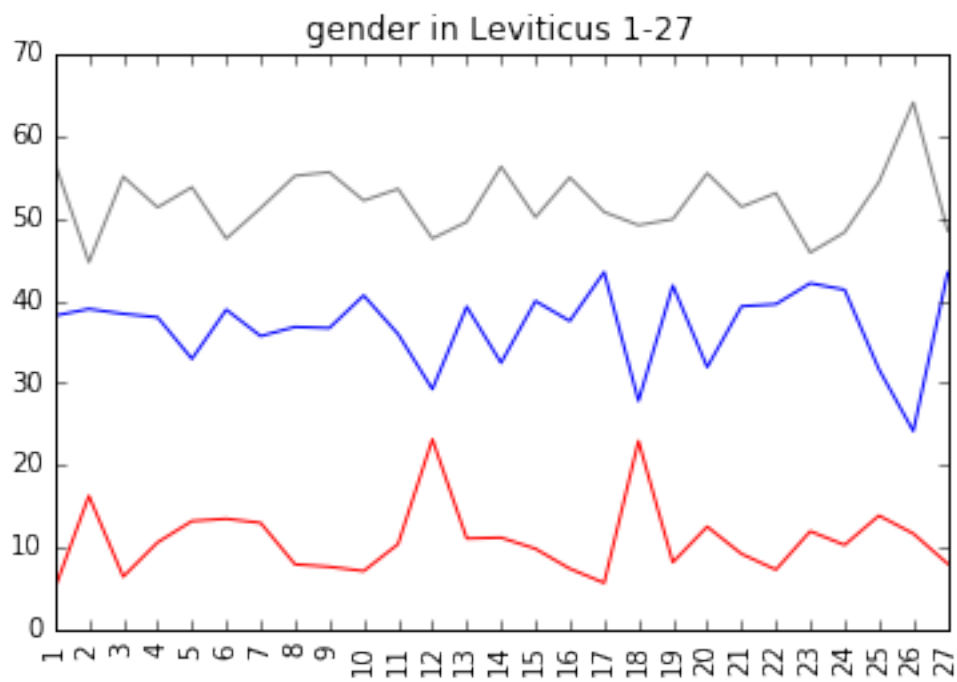
```
def atAGlance(book, chapter):
    words = L.d(T.nodeFromSection((book, chapter)), otype='word')
    freqs = collections.Counter()
    for w in words:
        if doingHebrew():
            lexeme = L.u(w, otype='lex')[0]
            freqs[F.gloss.v(lexeme)] += 1
        else:
            freqs[F.UnicodeLemma.v(w)] += 1
    for (gloss, freq) in sorted(freqs.items(), key=lambda x: (-x[1], x[0])):
        print('{:>3} {}'.format(freq, gloss))
```

In [30]:

```
def inDepth(book, chapter):
    chapterNode = T.nodeFromSection((book, chapter))
    verseNodes = L.d(chapterNode, otype='verse')
    for verseNode in verseNodes:
        words = L.d(verseNode, otype='word')
        print('{}: {}'.format(T.sectionFromNode(verseNode)[2], T.text(words)))
```

In [31]:

```
genderBias('Leviticus')
```



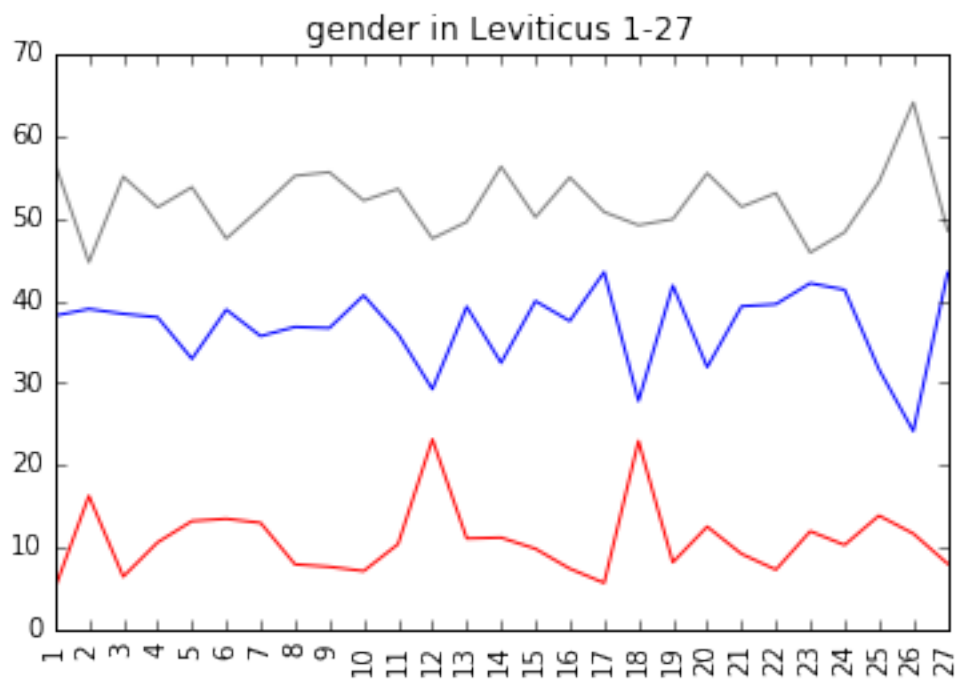
In [32]:

```
atAGlance('Leviticus', 18)
```

31 and
31 not
26 to
24 nakedness
21 the
18 <object marker>
17 uncover
14 in
11 woman
10 daughter
10 make
10 she
9 <relative>
9 father
8 be unclean
8 i
8 whole
7 YHWH
7 earth
6 abomination

In [33]:

```
genderBias('Leviticus')
```



In [34]:

```
atAGlance('Leviticus', 26)
```

```
132 and
54 <object marker>
49 in
40 to
32 the
23 earth
19 not
14 give
13 be hostile
11 even
11 i
10 from
10 sin
10 upon
10 walk
8 be
8 covenant
8 if
7 be desolate
7
```

In [35]:

```
inDepth('Leviticus', 26)
```

לֹא־תַעֲשׂוּ לָכֶם אֱלִילִים וּפָסָל וּמִצֵּבָה לֹא־תִקְיִמוּ לָכֶם וְאִבֹּן מִשְׁפִּית לֹא תִתְּנוּ בְּאַרְצְכֶם לֵה: שְׂתַחֲוֹת עָלֶיהָ כִּי אֲנִי יְהוָה אֱלֹהֵיכֶם: 2: אֶת־שַׁבְּתֹתַי תִּשְׁמְרוּ וּמִקֻּדְשֵׁי תִירָאוּ אֲנִי יְהוָה: ס 3: אִם־בִּחְקֹתַי תֵּלְכוּ וְאֶת־מִצְוֹתַי תִּשְׁמְרוּ וַעֲשִׂיתֶם אֹתָם: 4: וְנָתַתִּי גִשְׁמִיכֶם בְּעֵתָם וְנָתַנָּה הָאָרֶץ יְבוּלָה וְעַץ הַשָּׁדֶה יִתֶּן פֵּרוֹ: 5: וְהִשִּׁיג לָכֶם דִּישׁ אֶת־בָּצִיר וּבָצִיר יִשָּׂיג אֶת־זֶרַע וְאִכְלֹתֶם לַחֲמֻכָּם לְשִׁבְעַת יָמִים וְיִשְׁבַּתֶּם לְבָטָח ב: ארצכם: 6: וְנָתַתִּי שָׁלוֹם בְּאָרֶץ וּשְׂכֻבְתֶּם וְאִין מִחֲרִיד וְהִשְׁבַּתִּי חַיָּה וְעָה מִן־הָאָרֶץ וְחָרֹב לֹא־תֵעָבֹר בְּאַרְצְכֶם: 7: וְרִדְפְתֶּם אֶת־אִיְבֵיכֶם וְנָפְלוּ לִפְנֵיכֶם לְחָרֹב: 8: וְרִדְפוּ מִכָּם חֲמֹשֶׁה מֵאָה וּמֵאָה מִכָּם רַבְּבָה יִרְדְּפוּ וְנָפְלוּ אִיְבֵיכֶם לִפְנֵיכֶם לְחָרֹב: 9: וּפָנִיתִי אֲלֵיכֶם וְהִפְרִיתִי אֹתְכֶם וְהִרְבִּיתִי אֹתְכֶם וְהִקִּימֹתִי אֶת־בְּרִיתִי אִתְּכֶם: 10: וְאִכְלֹתֶם יֶשֶׁן נוֹשָׁן וְיֶשֶׁן מִפְּנֵי חֹדֶשׁ תּוֹצִיאוּ: 11: וְנָתַתִּי מִשְׁכָּנִי בְּתוֹכְכֶם וְלֹא־תִגְעַל נַפְשִׁי אִתְּכֶם: 12: וְהִתְהַלַּכְתִּי בְּתוֹכְכֶם וְהִיִּיתִי לָכֶם לֵאלֹהִים וְאַתֶּם תִּהְיוּ־לִי לְעָם: 13: אֲנִי יְהוָה אֱלֹהֵיכֶם אֲשֶׁר הוֹצֵאתִי אֹתְכֶם מֵאֶרֶץ מִצְרַיִם מֵהֵיטֵר לָהֶם עֲבָדִים וְאֲשַׁבֵּר מַטֵּה עַלְכֶּם וְאוֹלֶךְ אֹתְכֶם קוֹמָמִיּוֹת: פ 14: וְאִם־לֹא תִשְׁמָעוּ לִי וְלֹא תַעֲשׂוּ אֶת כָּל־הַמִּצְוֹת הָאֵלֶּה: 15: וְאִם־בִּחְקֹתַי תִּמָּאֲסוּ וְאִם אֶת־מִשְׁפָּטִי תִגְעַל נַפְשְׁכֶם לְבַלְתִּי עֲשׂוֹת אֶת־כָּל־מִצְוֹתַי לְהִשְׁתַּחֲוֹת לְאֱלֹהִים אֲחֵרִים:

Man, woman and thing

In [36]:

```
doGreek()
```

```
TF_G.load('UnicodeLemma', add=True)
```

0.00s loading features ...
0.00s All additional features loaded - for details use loadLog()

The Greek genders

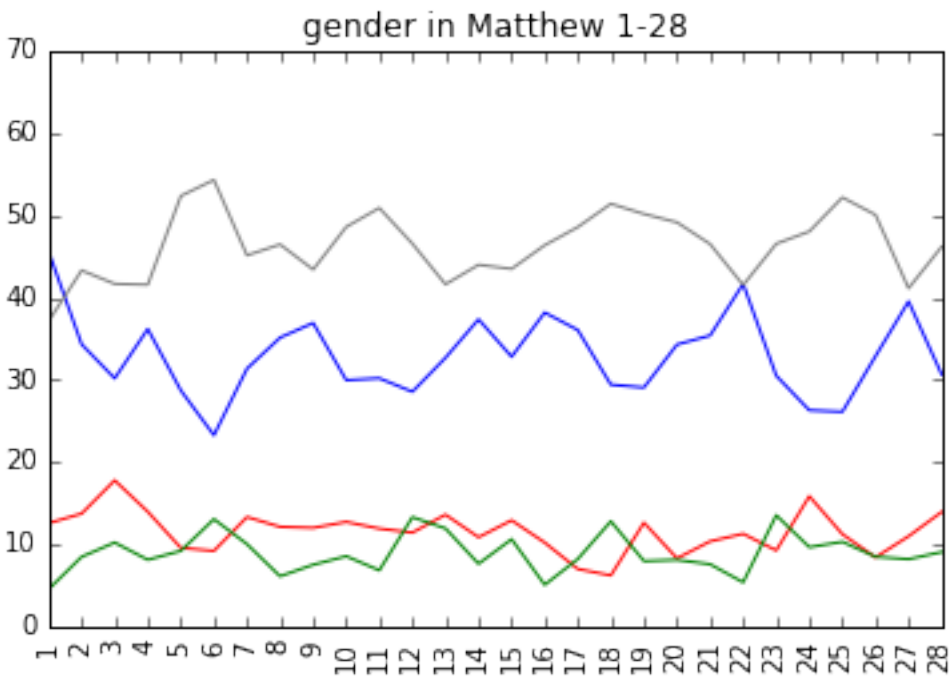
In [37]:

```
getGenders()
```

Out[37]:
{None, 'Neuter', 'Feminine', 'Masculine'}

In [38]:

```
genderBias('Matthew')
```



In [39]:

```
atAGlance('Matthew', 24)
```

- 139 ὁ
- 57 καί
- 23 αὐτός
- 16 εἰμί
- 16 ἐν
- 15 δέ
- 15 οὐ
- 14 μή
- 13 σύ
- 12 λέγω
- 11 ἐπί
- 10 ἡμέρα
- 9 οὗτος
- 9 πᾶς
- 9 τότε
- 9 ἐκεῖνος
- 8 γάρ
- 7 γίνομαι
- 7 οὐρανός
- 7 πάλαι

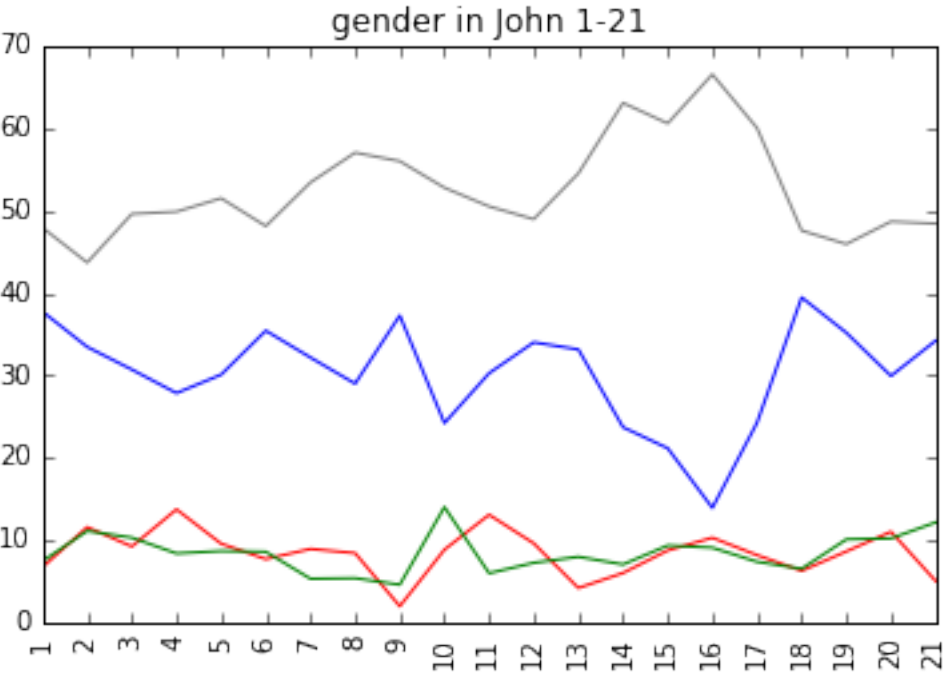
In [40]:

```
inDepth('Matthew', 24)
```

1: Καὶ ὁ Ἰησοῦς ἐξελθὼν ἀπὸ τοῦ ἱεροῦ ἐπορεύετο, καὶ προσῆλθον οἱ μαθηταὶ αὐτοῦ ἐπιδεῖξαι αὐτῷ τὰς οἰκοδομὰς τοῦ ἱεροῦ.
2: δὲ ὁ ἀποκριθεὶς εἶπεν αὐτοῖς· Οὐ βλέπετε ταῦτα πάντα; ἀμὴν λέγω ὑμῖν, οὐ μὴ ἀφεθῇ ὧδε ἐπὶ λίθον λίθος ὃς οὐ καταλυθήσεται.
3: δὲ Καθημένου αὐτοῦ ἐπὶ τοῦ Ὄρους τῶν Ἐλαιῶν προσῆλθον αὐτῷ οἱ μαθηταὶ κατ' ἰδίαν λέγοντες· Εἰπὸν ἡμῖν πότε ταῦτα ἔσται, καὶ τί τὸ σημεῖον τῆς σῆς παρουσίας καὶ συντελείας τοῦ αἰῶνος.
4: καὶ ἀποκριθεὶς ὁ Ἰησοῦς εἶπεν αὐτοῖς· Βλέπετε μὴ τις ὑμᾶς πλανήσῃ.
5: γὰρ πολλοὶ ἐλεύσονται ἐπὶ τῷ ὀνόματί μου λέγοντες· Ἐγὼ εἰμι ὁ χριστός, καὶ πολλοὺς πλανήσουσιν.
6: δὲ μελλήσετε ἀκούειν πολέμους καὶ ἀκοὰς πολέμων· ὁρᾶτε, μὴ θροεῖσθε· γὰρ δεῖ γενέσθαι, ἀλλ' οὕπω ἔστιν τὸ τέλος.
7: γὰρ ἐγερθήσεται ἔθνος ἐπὶ ἔθνος καὶ βασιλεία ἐπὶ βασιλείαν, καὶ ἔσονται λιμοὶ καὶ σεισμοὶ κατὰ τόπους.
8: δὲ πάντα ταῦτα ἀρχὴ ὠδίνων.
9: Τότε παραδώσουσιν ὑμᾶς εἰς θλίψιν καὶ ἀποκτενοῦσιν ὑμᾶς, καὶ ἔσεσθε μισούμενοι ὑπὸ πάντων τῶν ἐθνῶν διὰ τὸ ὄνομά μου.
10: καὶ τότε σκανδαλισθήσονται πολλοὶ καὶ ἀλλήλους παραδώσουσιν καὶ μισήσουσιν ἀλλήλους.
11: καὶ πολλοὶ ψευδοπροφῆται ἐκσταθήσονται καὶ πλανήσουσιν πολλοὺς.

In [41]:

```
genderBias('John')
```



In [42]:

```
atAGlance('John', 16)
```

```
52 ὁ
40 σύ
34 καί
26 ἐγώ
21 ὅτι
16 λέγω
14 οὐ
13 οὗτος
12 ἐν
11 πατήρ
11 ἔρχομαι
10 αὐτός
10 λαλέω
9 δέ
9 περί
9 ἀλλά
8 κόσμος
8 ἵνα
7 εἰμί
7 ἡμεῖς
```

Six days of work (creating data)

Semantic plurals in the letter of Jude.

Let's get all nominal phrases.

In [43]:

```
doGreek()
TF_G.load('Cat', add=True)
```

```
0.00s loading features ...
|      0.18s B Cat
ric-data/greek/sblgnt
0.19s All additional features loaded - for details use loadLog()
```

In [44]:

```
bookNode = T.nodeFromSection(('Jude',))
phraseNodes = L.d(bookNode, otype='phrase')
NPs = [p for p in phraseNodes if F.Cat.v(p) == 'np']

print('{} NPs in Jude'.format(len(NPs)))
```

```
357 NPs in Jude
```

Generate a data entry form

Export this data as CSV so that experts can fill in a new feature: *semantically plural*.

In [45]:

```
enrichFile = 'np.csv'
enrichedFile = 'np-enriched.csv'

with open(enrichFile, 'w') as f:
    fieldNames = ['passage', 'node', 'phrase', 'semantic plural', 'sentence']
    f.write('{}\n'.format('\t'.join(fieldNames)))
    for np in NPs:
        sn = L.u(np, otype='sentence')[0]
        sentence = L.d(sn, otype='word')
        phrase = L.d(np, otype='word')
        fields = [
            '{} {}:{}'.format(*T.sectionFromNode(np)),
            str(np),
            T.text(phrase),
            '',
            T.text(sentence),
        ]
        f.write('{}\n'.format('\t'.join(fields)))
```

In [46]:

```
dataFrame = pandas.read_csv(enrichFile, sep='\t')
dataFrame.head(100)
```

Out[46]:

	passage	node	phrase	semantic plural	sentence
0	Jude 1:1	400492	Ἰούδας Ἰησοῦ Χριστοῦ δοῦλος, δὲ ἀδελφὸς Ἰακώβου,	NaN	Ἰούδας Ἰησοῦ Χριστοῦ δοῦλος, δὲ ἀδελφὸς Ἰακώβο...
1	Jude 1:1	400493	Ἰούδας	NaN	Ἰούδας Ἰησοῦ Χριστοῦ δοῦλος, δὲ ἀδελφὸς Ἰακώβο...
2	Jude 1:1	400494	Ἰησοῦ Χριστοῦ δοῦλος, δὲ ἀδελφὸς Ἰακώβου,	NaN	Ἰούδας Ἰησοῦ Χριστοῦ δοῦλος, δὲ ἀδελφὸς Ἰακώβο...
3	Jude 1:1	400495	Ἰησοῦ Χριστοῦ δοῦλος,	NaN	Ἰούδας Ἰησοῦ Χριστοῦ δοῦλος, δὲ ἀδελφὸς Ἰακώβο...
4	Jude 1:1	400496	Ἰησοῦ Χριστοῦ	NaN	Ἰούδας Ἰησοῦ Χριστοῦ δοῦλος, δὲ ἀδελφὸς Ἰακώβο...
5	Jude 1:1	400497	Ἰησοῦ	NaN	Ἰούδας Ἰησοῦ Χριστοῦ δοῦλος, δὲ ἀδελφὸς Ἰακώβο...

Read the data enrichments

In [47]:

```
semNumber = dict()

with open(enrichedFile) as f:
    for (i, line) in enumerate(f):
        if i == 0: continue                # header row

        fields = line.rstrip('\n').split(';')
        value = fields[3]
        if value == '': continue          # no data entered

        node = int(fields[1])
        semNumber[node] = value
```

In [48]:

```
for p in sorted(semNumber):
    print('{} => {}'.format(p, semNumber[p]))
```

```
400518 => p
400523 => p
400536 => p
400538 => s
400544 => s
400549 => p
400553 => p
400578 => s
400579 => s
400595 => p
```

Save the new feature as a text-fabric file

In [49]:

```
metaData = dict(
    semNumber=dict(
        valueType='str',
        source='Semantic plurality training set',
        author='J.S. Bach, Leipzig',
    ),
)
TF_G = Fabric(locations='.', modules='semantic')
```

This is Text-Fabric 2.3.7

Api reference : <https://github.com/ETCBC/text-fabric/wiki/Api>

(<https://github.com/ETCBC/text-fabric/wiki/Api>)

Tutorial : [https://github.com/ETCBC/text-fabric/blob/master/docs/](https://github.com/ETCBC/text-fabric/blob/master/docs/tutorial.ipynb)

[tutorial.ipynb](https://github.com/ETCBC/text-fabric/blob/master/docs/tutorial.ipynb) ([https://github.com/ETCBC/text-fabric/blob/master/docs/](https://github.com/ETCBC/text-fabric/blob/master/docs/tutorial.ipynb)

[tutorial.ipynb](https://github.com/ETCBC/text-fabric/blob/master/docs/tutorial.ipynb))

Data sources : <https://github.com/ETCBC/text-fabric-data>

(<https://github.com/ETCBC/text-fabric-data>)

Data docs : <https://etcbc.github.io/text-fabric-data>

(<https://etcbc.github.io/text-fabric-data>)

Shebanq docs : <https://shebanq.ancient-data.org/text>

(<https://shebanq.ancient-data.org/text>)

Slack team : <https://shebanq.slack.com/signup>

(<https://shebanq.slack.com/signup>)

Questions? Ask shebanq@ancient-data.org for an invite to Slack

1 features found and 0 ignored

0.00s Grid feature "otype" not found in
/Users/dirk/github/text-fabric/presentations/Leipzig2017-07-06/semantic

0.00s Grid feature "oslots" not found in
/Users/dirk/github/text-fabric/presentations/Leipzig2017-07-06/semantic

0.01s Grid feature "otext" not found. Working without Text-API

In [50]:

```
TF_G.save(
    nodeFeatures=dict(semNumber=semNumber),
    metaData=metaData,
)
```

0.00s Exporting 1 node and 0 edge and 0 config features to /Users/dirk/github/text-fabric/presentations/Leipzig2017-07-06/semantic:

| 0.00s T semNumber to /Users/dirk/github/text-fabric/presentations/Leipzig2017-07-06/semantic

0.01s Exported 1 node features and 0 edge features and 0 config features to /Users/dirk/github/text-fabric/presentations/Leipzig2017-07-06/semantic

Check

In [51]:

```
!cat semantic/semNumber.tf
```

```
@node
@author=J.S. Bach, Leipzig
@source=Semantic plurality training set
@valueType=str
@writtenBy=Text-Fabric
@dateWritten=2017-07-19T12:19:42Z
```

```
400518  p
400523  p
400536  p
400538  s
400544  s
400549  p
400553  p
400578  s
s
400595  p
```

Use the new feature

In [52]:

```
LOCATIONS = [
    '~/Downloads/text-fabric-data',
    '~/text-fabric-data',
    '~/github/text-fabric-data',
    '/mnt/shared/text-fabric-data',
]

TF_G = Fabric(
    locations=LOCATIONS+[ '.' ],
    modules=['greek/sblgnt', 'semantic'],
)
```

This is Text-Fabric 2.3.7

Api reference : <https://github.com/ETCBC/text-fabric/wiki/Api>

(<https://github.com/ETCBC/text-fabric/wiki/Api>)

Tutorial : [https://github.com/ETCBC/text-fabric/blob/master/docs/](https://github.com/ETCBC/text-fabric/blob/master/docs/tutorial.ipynb)

tutorial.ipynb ([https://github.com/ETCBC/text-fabric/blob/master/docs/](https://github.com/ETCBC/text-fabric/blob/master/docs/tutorial.ipynb)

Data sources : <https://github.com/ETCBC/text-fabric-data>

(<https://github.com/ETCBC/text-fabric-data>)

Data docs : <https://etcbc.github.io/text-fabric-data>

(<https://etcbc.github.io/text-fabric-data>)

Shebanq docs : <https://shebanq.ancient-data.org/text>

(<https://shebanq.ancient-data.org/text>)

Slack team : <https://shebanq.slack.com/signup>

(<https://shebanq.slack.com/signup>)

Questions? Ask shebanq@ancient-data.org for an invite to Slack

64 features found and 0 ignored

In [53]:

```
apiG = TF_G.load('Number semNumber')
doGreek()

0.00s loading features ...
|      0.04s B Number          from /Users/dirk/github/text-fab
ric-data/greek/sblgnt
|      0.01s T semNumber      from /Users/dirk/github/text-fab
ric/presentations/Leipzig2017-07-06/semantic
|      0.00s Feature overview: 61 for nodes; 2 for edges; 1 configs;
7 computed
1.40s All features loaded/computed - for details use loadLog()
```

Observe it in action

In [54]:

```
for np in NPs:
    semNumber = F.semNumber.v(np)
    if not semNumber: continue
    words = L.d(np, otype='word')
    print('NP {}: semantically "{}", words marked as {}'.format(
        np,
        semNumber,
        ' '.join(F.Number.v(w) for w in words if F.Number.v(w)),
    ))
```

```
NP 400518: semantically "p", words marked as Singular Singular Singula
r
NP 400523: semantically "p", words marked as Plural
NP 400536: semantically "p", words marked as Plural
NP 400538: semantically "s", words marked as Singular
NP 400544: semantically "s", words marked as Singular Singular Plural
Plural Singular
NP 400549: semantically "p", words marked as Plural Plural
NP 400553: semantically "p", words marked as Plural Plural Plural Plur
al Singular Singular Singular Plural
NP 400578: semantically "s", words marked as Singular Singular Singula
r Singular Plural Singular Singular
NP 400579: semantically "s", words marked as Singular Singular Singula
r Plural Singular Singular
NP 400595: semantically "p", words marked as Plural
```

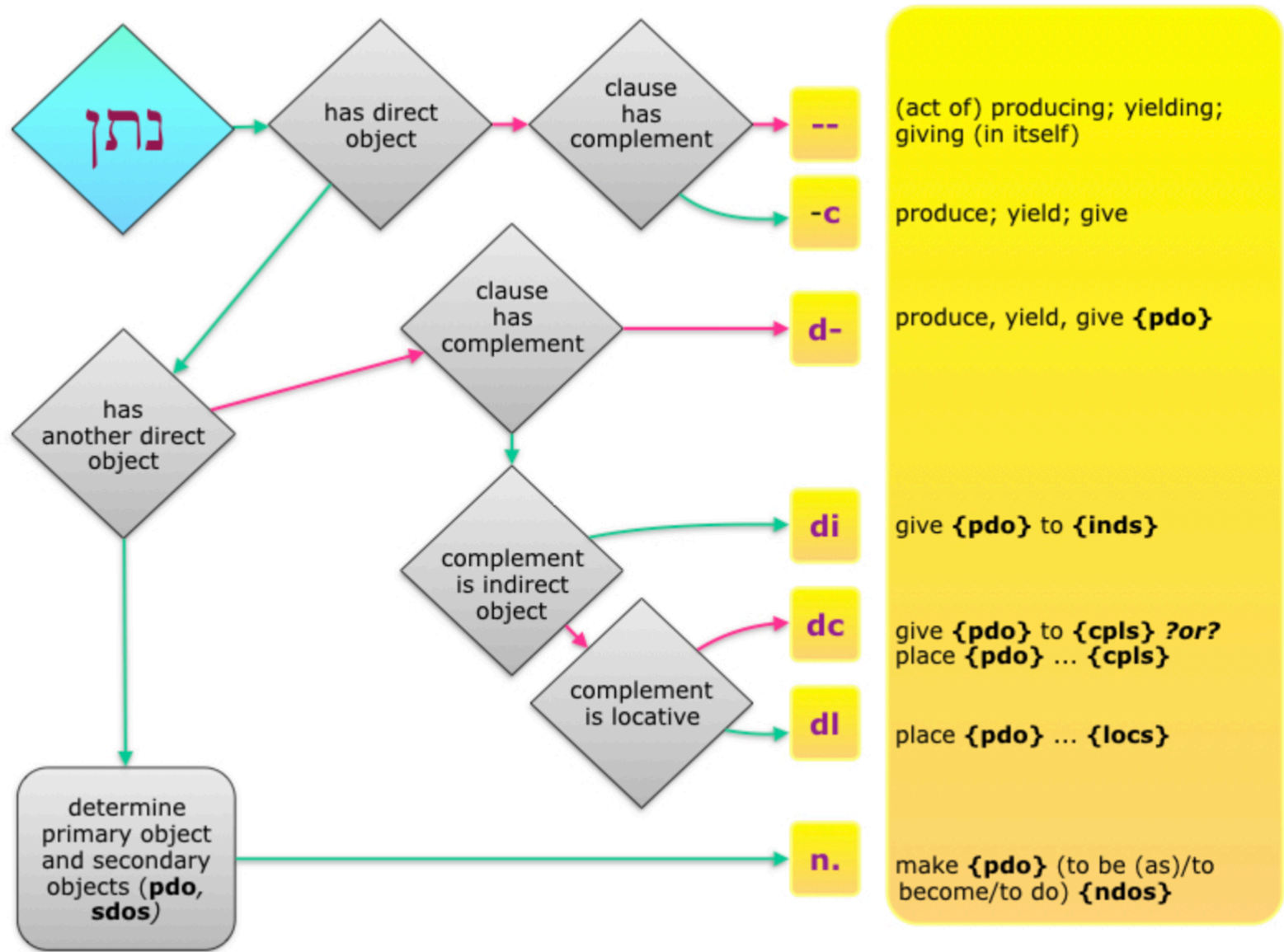
Sabbath

Have a look at the (un)finished work and see whether it is good.

Janet Dyk

Verbal valence flowchart.

As an example, this is a simplified flowchart for NTN in diagram form as we will implement it below.



Flowchart logic

Martijn Naaijer

Won a grassroots price for setting up a theology course based on SHEBANQ, Jupyter, and R. See [Python course here](#)

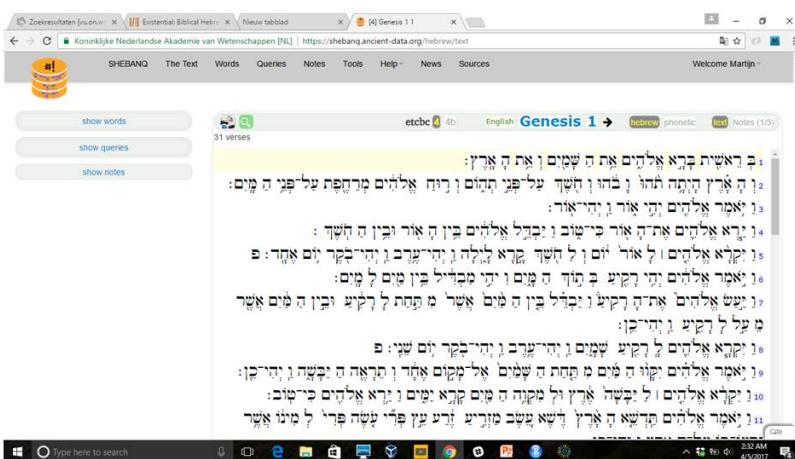
(https://shebanq.jove.surfsara.nl/user/dirk/notebooks/shared/martijn/Python_Course/Introduction_to_text_fabric)

Bijbelwetenschap als data science Theologisch onderwijs met Jupyter Notebook

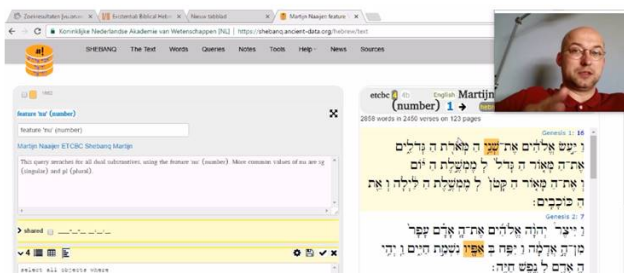
Martijn Naaijer
Faculteit der Godgeleerdheid

Van onderzoeksvraag

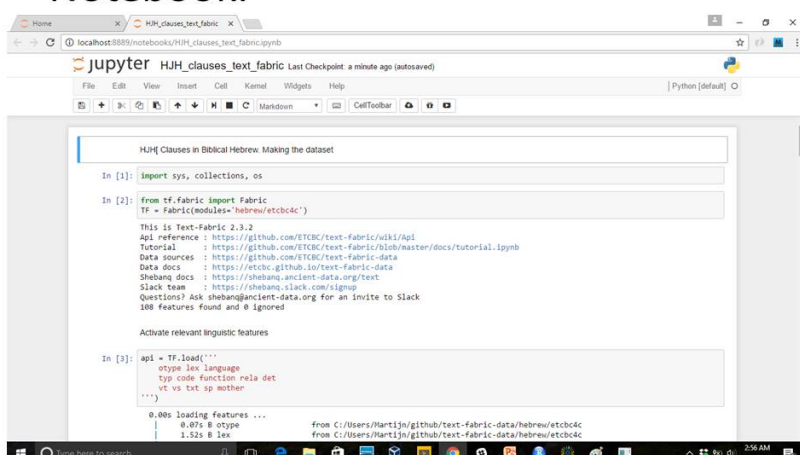
Studenten formuleren een onderzoeksvraag mbt een taalkundig probleem in het Bijbels Hebreeuws.



Via kennisclips leren ze queries te maken in de ETCBC database van de Hebreeuwse Bijbel op onze website shebanq.ancient-data.org



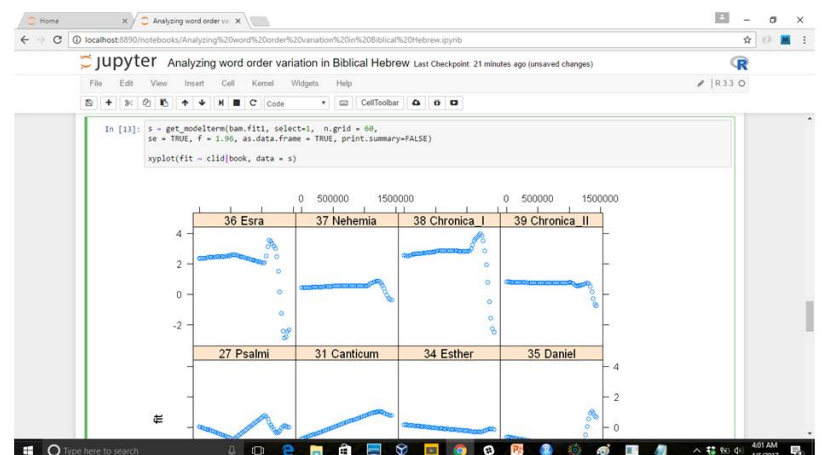
Ook gaan ze datasets maken mbv de Python package text-fabric in Jupyter Notebook.



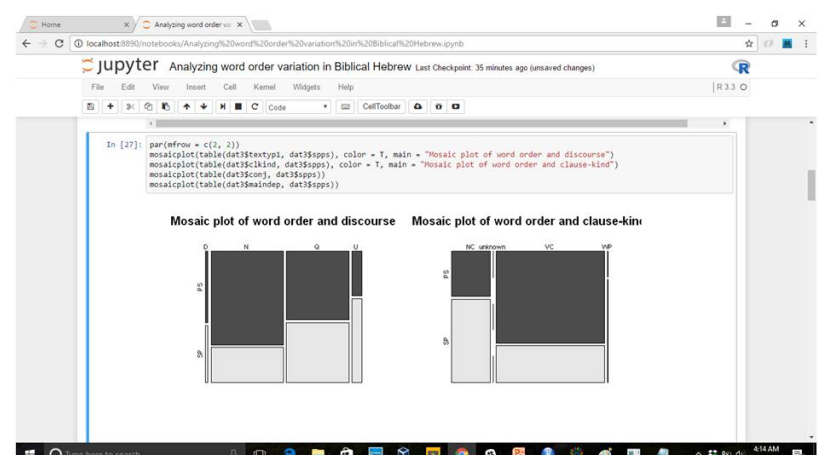
Naar analyse en verslag

Analyses van de gemaakte datasets worden gedaan met R in Jupyter Notebook en hierin wordt ook een compleet onderzoeksverslag gemaakt met Markdown.

Studenten zetten hun verslag op github.com en beoordelen elkaars verslagen.

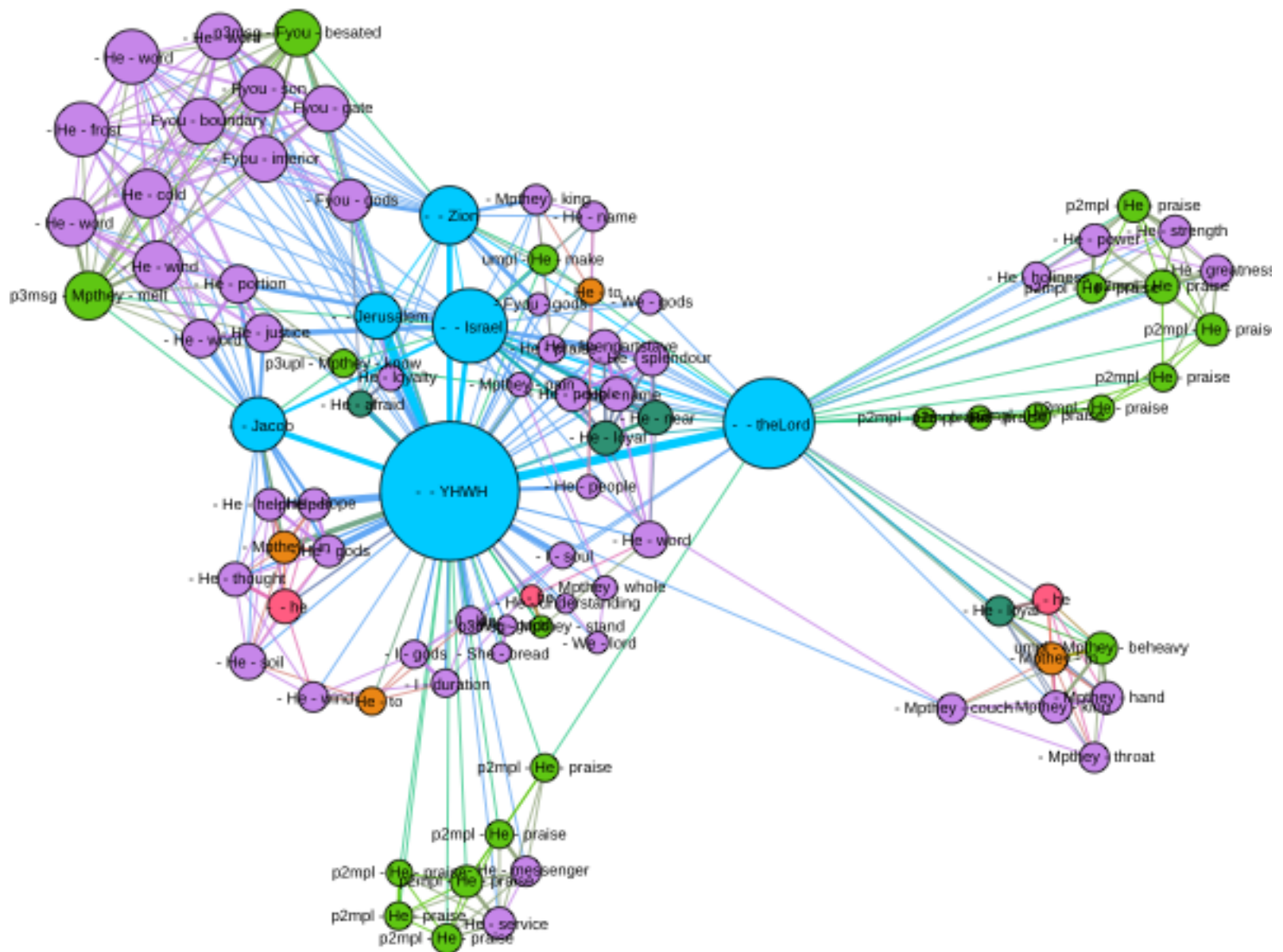


Alle leermiddelen (video's, voorbeeldqueries, Python en R-scripts) zijn Open Source beschikbaar via shebanq.ancient-data.org en github.com



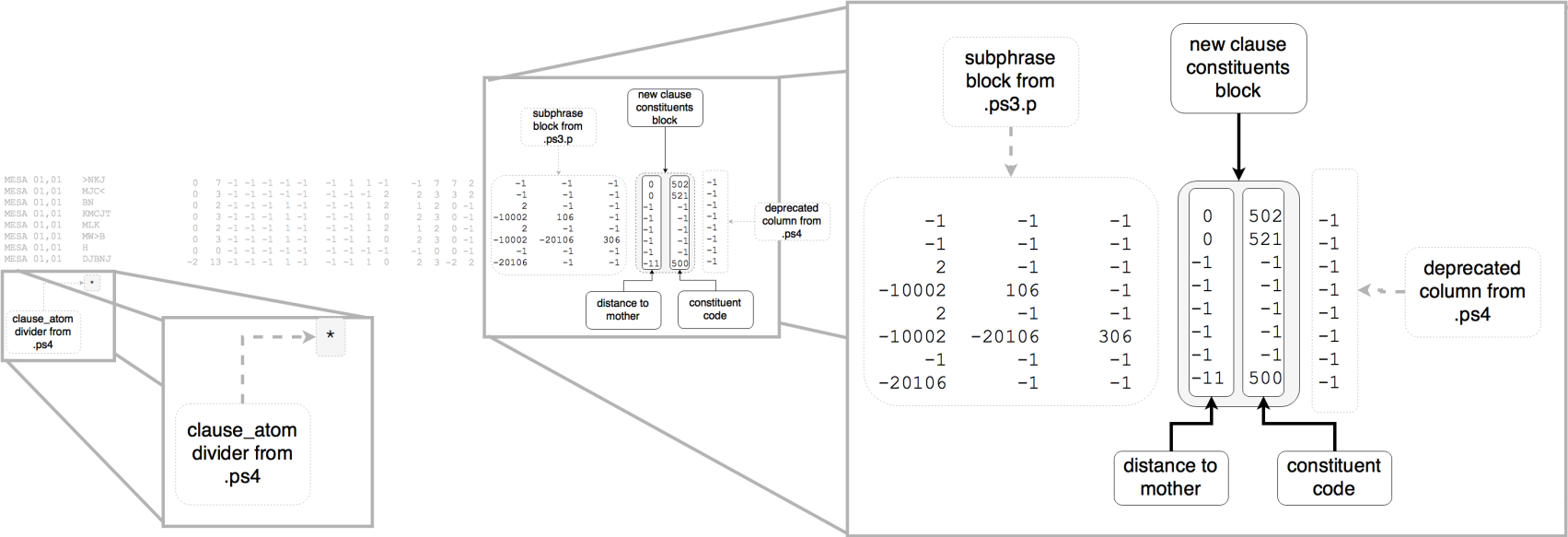
Christiaan Erwich

Tries to track who is who in the Psalms, and is deeply into graph visualization.



Cody Kingham

Helped to convert the SBL Greek New Testament to the Text-Fabric format. Tries to explain to the world (<http://www.codykingham.com/etcbc/datacreation>) how the ETCBC encoded the Hebrew Bible during a 40 year long struggle with computers.



Dirk Roorda

Tries to recombine everything.

Phonetic transcription of Hebrew (<https://rawgit.com/ETCBC/text-fabric/master/phono/phonoTf.html>)

1574 tests; 0 skipped; 0 failed; 1574 passed of which 1377 exactly.

v	verse	etcbc	hebrew	lexical	phono	expected	comment
1	Numeri 22:20 357777	B.I.L:<@M02	בִּלְעָם	nmpr sm,	bilʕām	bilʕāmʾ	on the basis of other occurrences
2	Numeri 22:5 357777	B.I.L:<@74M	בִּלְעָם	nmpr sm,	bilʕʾām		by the rules
3	Numeri 22:10 357777	B.I.L:<@73M	בִּלְעָם	nmpr sm,	bilʕ,ām		by the rules
4	Zephania 3:19 357777	B.@C:T.@75M00	בִּשְׁתָּם:	subs sf,+@m	boštʾām		by the rules
5	Jesaja 61:7 357777	B.@C:T.:KEM03	בִּשְׁתָּכֶם	subs sf,+:kem	boštʔk,em		by the rules
6	Deuteronomium 7:26 357777	B.;;JTE80K@	בֵּיתָךְ	subs sm,+ek@	bêtʔekā		by the rules

Parallel passages (<https://shebanq.ancient-data.org/shebanq/static/docs/tools/parallel/parallels.html>)

See it in action on SHEBANQ: etcbc4b Genesis 10:1 (https://shebanq.ancient-data.org/hebrew/text?qactive=hlcustom&qsel_one=grey&qpub=x&qget=x&wactive=hlcustom&wsel_one=gray&wpub=x&wget=x&nacti)

[show words](#)[show queries](#)[hide notes](#)

”

[D Roorda.. crossref](#)[D Roorda.. exegese](#)

etcbc 4 4b English Genesis 10 Hebrew phonetic text Notes (1/3)

32 verses

1

תּוֹלַדַּת בְּנֵי־נֹחַ שֵׁם תָּם וְיֶפֶת אֱלֹהִים

PreC 3 Subj 2 Conj 1

chapter diff with Chronica_I 1

Dirk Roorda

אֶתֶר 5 בָּנִים 4 לָהֶם 3 יִנְלְדוּ 2

Time Subj Cmpl Pred 1 Conj

הַמִּבּוּל:

2

נֹחַר וּמִגּוֹג וּמִדֵּי וַיֵּן וְתָבֵל וּמִשְׁדֵּךְ בְּנֵי יֶפֶת

PreC 2 Subj 1

«Chronica_I 1:5 ~100%»

Dirk Roorda

וְתִירָם:

all variants (clique 5)

Dirk Roorda

Stand-off markup for changing sources

(it is not a nightmare)

Versioning (<https://github.com/ETCBC/text-fabric/blob/master/Versions/etcbc-versions.ipynb>)

Thanks

dirk.roorda@dans.knaw.nl
(<mailto:dirk.roorda@dans.knaw.nl>)

Linguistic Annotation and Philology

Workshop (<http://www.dh.uni-leipzig.de/wo/laphw/>)

Leipzig, July 6-7, 2017

([https://dans.knaw.nl/en/front-page?](https://dans.knaw.nl/en/front-page?set_language=en)

[set_language=en](https://dans.knaw.nl/en/front-page?set_language=en))



google github text-fabric wiki (<https://github.com/ETCBC/text-fabric/wiki>)

Data Archiving and Networked Services

DANS

