Introduction to Web Science

Assignment 6

Prof. Dr. Steffen Staab
staab@uni-koblenz.de

René Pickhardt
rpickhardt@uni-koblenz.de

Korok Sengupta
koroksengupta@uni-koblenz.de

Institute of Web Science and Technologies
Department of Computer Science
University of Koblenz-Landau

Submission until: December 6, 2016, 10:00 a.m.
Tutorial on: December 9, 2016, 12:00 p.m.

Please look at the lessons 1) **Simple descriptive text models** & 2) **Advanced descriptive text models**

For all the assignment questions that require you to write code, make sure to include the code in the answer sheet, along with a separate python file. Where screen shots are required, please add them in the answers directly and not as separate files.

Team Name: Bravo - Shriharsh Ambhore, Kandhasamy Rajasekaran, Daniel Akbari

1 Digging deeper into Norms (10 points)

You have been introduced to the concept of a norm and have seen that the uniform norm $||\cdot||_\infty$ fullfills all three axioms of a norm which are:

1. Positiv definite
2. Homogeneous
3. Triangle inequality

Recall that for a function $f : M \longrightarrow \mathbb{R}$ with M being a finite set[1] we have defined the L_1-norm of f as:

$$||f||_1 := \sum_{x \in M} |f(x)| \tag{1}$$

In this exercise you should

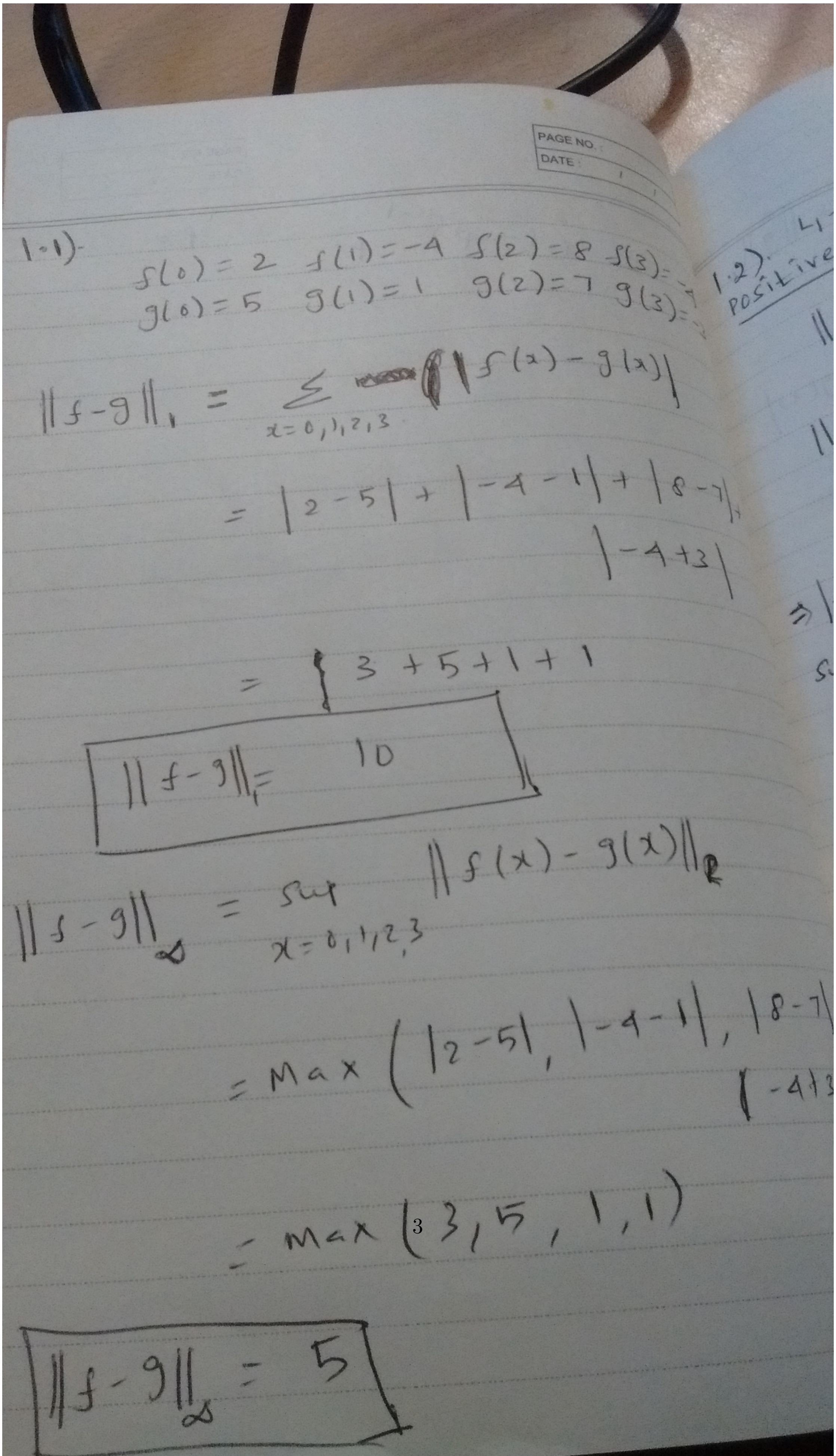
1. calculate $||f-g||_1$ and $||f-g||_\infty$ for the functions f and g that are defined as
   - $f(0) = 2, f(1) = -4, f(2) = 8, f(3) = -4$ and
   - $g(0) = 5, f(1) = 1, g(2) = 7, g(3) = -3$
2. proof that all three axioms for norms hold for the L_1-norm.

1.1 Hints:

1. The proofs work in a very similar fashion to those from the uniform norm that was depicted in the videos.
2. You can expect that the proofs for each property also will be ”three-liners”.
3. Both parts of this exercise are meant to practice proper and clean mathematical notation as this is very helpfull when reading and understanding research papers. Discuss in your study group not only the logics of the calculation and the proof (before submission) but try to emphasize on the question whether your submission is able to communicate exactly what you are doing.

[1] You could for example think of the function measuring the frequency of a word depening on its rank.

Figure 1: Question 1 : Page 1



PAGE NO.
DATE

1.1)

$f(0) = 2 \quad f(1) = -4 \quad f(2) = 8 \quad f(3) =$

$g(0) = 5 \quad g(1) = 1 \quad g(2) = 7 \quad g(3) =$

$$\|f-g\|_1 = \sum_{x=0,1,2,3} |f(x) - g(x)|$$

$$= |2-5| + |-4-1| + |8-7| + |-4+3|$$

$$= 3+5+1+1$$

$$\boxed{\|f-g\|_1 = 10}$$

$$\|f-g\|_\infty = \sup_{x=0,1,2,3} \|f(x)-g(x)\|_R$$

$$= \mathrm{Max}\left(|2-5|, |-4-1|, |8-7|, |-4+3\right.$$

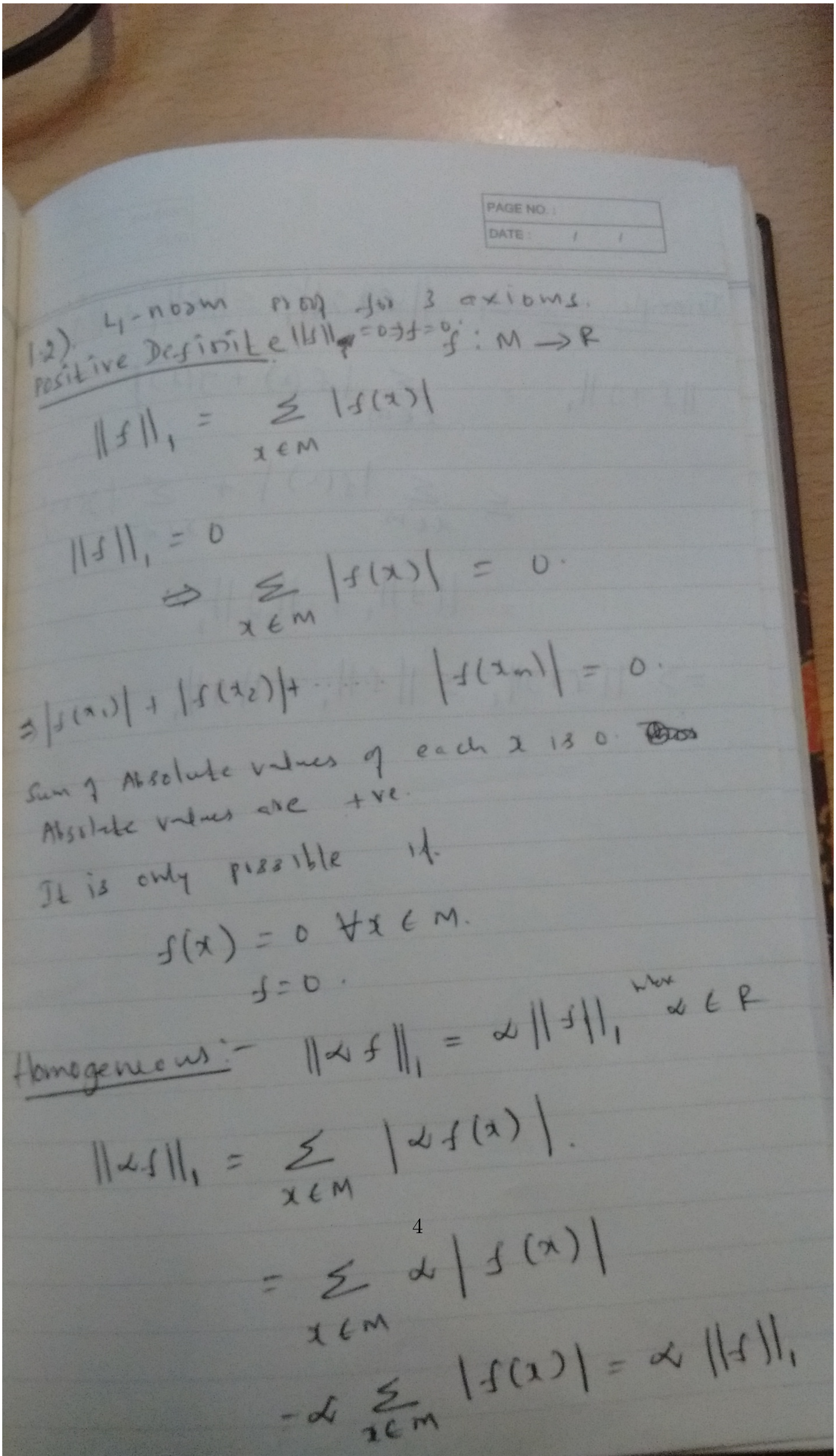
$$= \max(3, 5, 1, 1)$$

$$\boxed{\|f-g\|_\infty = 5}$$

1.2) Positive

L1

Figure 2: Question 1 : Page 2



PAGE NO. :

DATE : / /

1.2) L_1-norm proof for 3 axioms.

Positive Definite $\|f\|=0 \Rightarrow f=0$; $f : M \rightarrow R$

$$\|f\|_1 = \sum_{x \in M} |f(x)|$$

$$\|f\|_1 = 0$$

$$\Rightarrow \sum_{x \in M} |f(x)| = 0.$$

$$\Rightarrow |f(x_1)| + |f(x_2)| + \cdots |f(x_m)| = 0.$$

Sum of Absolute values of each x is 0.

Absolute values are +ve.

It is only possible if.

$$f(x) = 0 \ \forall x \in M.$$

$$f = 0.$$

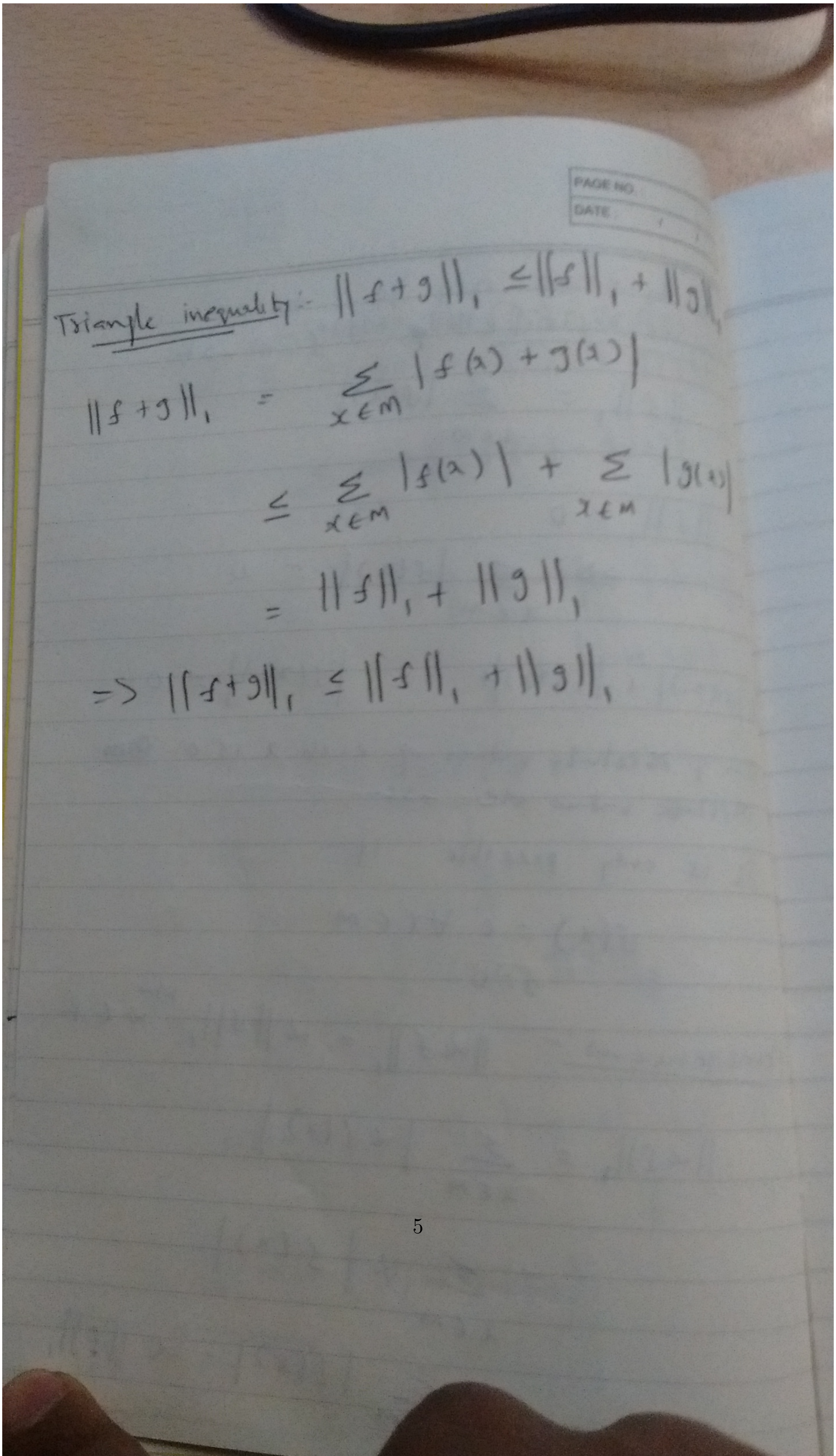
Homogeneous :- $\|\alpha f\|_1 = \alpha \|f\|_1$ where $\alpha \in R$

$$\|\alpha f\|_1 = \sum_{x \in M} |\alpha f(x)|.$$

$$= \sum_{x \in M} \alpha |f(x)|$$

$$= \alpha \sum_{x \in M} |f(x)| = \alpha \|f\|_1$$

Figure 3: Question 1 : Page 3



PAGE NO.

DATE

Triangle inequality: $\|f+g\|_1 \le \|f\|_1 + \|g\|_1$

$$\|f+g\|_1 = \sum_{x \in M} |f(x) + g(x)|$$

$$\le \sum_{x \in M} |f(x)| + \sum_{x \in M} |g(x)|$$

$$= \|f\|_1 + \|g\|_1$$

$$\Rightarrow \|f+g\|_1 \le \|f\|_1 + \|g\|_1$$

2 Coming up with a research hypothesis (12 points)

You can find all the text of the articles from Simple English Wikipedia at http://141.26.208.82/simple-20160801-1-article-per-line.zip each line contains one single article.

In this task we want you to be creative and do some research on this data set. The ultimate goal for this exercise is to practice the way of coming up with a research hypothesis and testable predictions.

In order to do this please **shortly**[2] answer the following questions:

1. What are some obervations about the data set that you can make? State at least three obervations.
2. Which of these observations make you curious and awaken your interest? Ask a question about why this pattern could occur.
3. Formulate up to three potentiel research hypothesis.
4. Take the most promesing hypothesis and develop testable predictions.
5. Explain how you would like to use the data set to test the prediction by means of descriptive statistics. Also explain how you would expect your outcome.

   (If you realize that the last two steps would not lead anywhere repeat with one of your other research hypothesis.)

2.1 Hints:

- The first question could already include some diagrams (from the lecture or ones that you did yourselves).
- In step 3 explain how each of your hypothesis is falsifiable.
- In the fifth step you could state something like: ”We expect to see two diagrams. The first one has ... on the x-axis and ... on the y-axis. The image should look like a ... The second diagram ...”. You could even draw a sketch of the diagram and explain how this would support or reject your testable hypothesis.

Answer: 2.1) From the first 6 articles

The subject of these articles are independent of each other They are grammatically well formed. Most of the words are made up of english alphabet They are short and will not take time to read the first word repeats in each article Punctuation are used in these articles

[2] Depending on the question shortly could mean one or two sentences or up to a thousand characters. We don’t want to give a harsh limit because we trust in you to be reasonable.

2.2) We are interested in the following observations

The subject of these articles are independent of each other It means the contributors are diverse (their field of interest). This means the whole set attracts readers with different taste

They are grammatically well formed. From this, we will understand the quality of simple english wikipedia.

Most of the words are made up of english alphabet It is Simple English Wikipedia. We want to know how much of 'English' is in it :-). Is it totally english or some combinations of other languages.

the first non function word repeats in the article - If the first non-function word gets repeated then it has high probability of being the topic of the article. This behavior conveys the idea that authors start with the topic first

Punctuation are used in these articles - It talks about the narrative of the content creator. It also refers to the importance he/she gives to the grammar.

2.3)

Observation : Most of the words are made up of english alphabet Hypothesis : More than 90 percent of characters in the simple english wikipedia articles are from english alphabets Falsifiable condition: If we find more than 10 percent of non english characters (a - z, A - Z) then it is falsified

Observation : the first word repeats in each article Hypothesis : More than 90 percent of simple english wikipedia is repeated article have their first non-functional word repeated Falsifiable condition: If we find more than 10 percent of articles have their non-functional word only once then it is falsified

Observation : Punctuation are used in these articles Hypothesis : All the articles in simple english wikipedia have atleast one punctuation mark Falsifiable condition: If we find at least one article without any punctuation then it is falsified

2.4) We take the following hypothesis 'More than 80 percent of characters in the simple english wikipedia articles are from english alphabets'

Testable predictions:

(Count the english alphabets / Count the total characters) $> = 0.9$

Assumptions : English alphabets - A to Z , a to z Characters excluded - spaces, tabs, new line and carriage return

2.5) We need all the articles of simple english wikipedia. We go through one by one and calculate the english alphabets count and total count. We check whether their ratios are satisfying our testable predictions. If it is greater than 0.9 then our hypothesis is true otherwise false.

The above is good enough to do it. But we wanted to use some graphs :-)

We will need a graph which has articles on x-axis, normalized english alphabets usage in y axis-

3 Statistical Validity (8 points)

In the above question, you were asked to formulate your hypothesis. In this one, you should follow your own defined roadmap from task 2 validate (or reject) your hypothesis.

3.1 Hints:

- In case feel uncomfortable to test one of the predictions from task 2 you can ”steal” one of the many hypothesis (and with them imlicitly associated testable predictions) or diagrams depicted from the lecture and reproduce it. However in that case you cannot expect to get the total amount of points for task 3.

We have counted the numbers of characters inside the whole file with exception of the spaces. We counted the number of english characters and divide it by total number of characters.

These are the result we got:

Words 79164090

Number of English Chars 74186792

Number of Non English Chars 5111382

Total Chars 79298174

Percentage 93.55422484255438

As we have 93.5 percent English characters ,we can validate our hypothesis, that more than 90 percent of it is English.

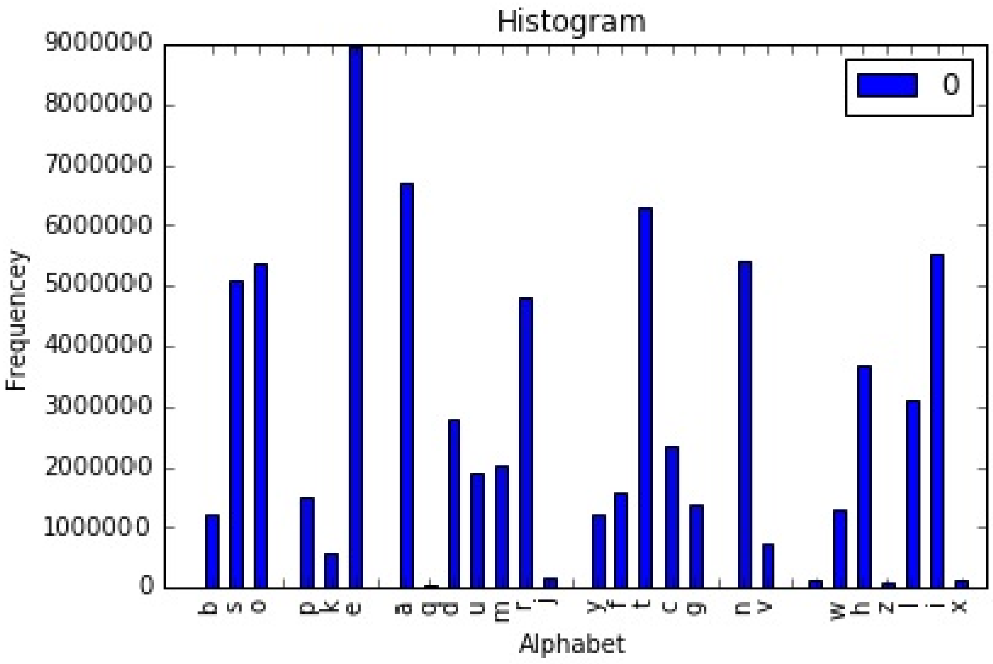
We also can see in the charts what are the frequency of each alphabet in our sample data. Which shows that ”e” is the most repeated character.

3.2 count.py

```

# -*- coding: utf-8 -*-
"""
Created on Mon Dec  5 17:42:14 2016
@author: Daniel
"""
import codecs
import matplotlib.pyplot as plt
import pandas
from collections import Counter
#import numpy as np

```

```
filename = "simple-20160801-1-article-per-line"

alphabet = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ"

num_lines = 0
num_words = 0
num_Enchars = 0
num_Othchars =0

x= list()
y= list()


with codecs.open(filename , encoding='utf-8') as file:
#with codecs.open(filename) as file:
    text = file.read()
    text= text.replace(" ", "")
    for line in text:
        words = line.split()
        num_words += len(words)
        if line.strip() in alphabet:

            num_Enchars += len(line)
            x.append(line.lower())

        else:
            num_Othchars +=len(line)
            y.append(line)

x.sort()
#print(x)
#print(y)

def Histogram(x):
    letter_counts = Counter(x)
    df = pandas.DataFrame.from_dict(letter_counts, orient='index')
    df.plot(kind='bar')

    #plt.hist(len(letter_counts),bins=5)
    plt.title('Histogram')
    plt.xlim(-2,32)
    plt.xlabel("Alphabet")
    plt.ylabel("Frequencey")
    plt.show()

def Bar(x) :

    performance = [9000,4500,0]
    plt.bar(x, performance, align='center', alpha=0.5)
    #plt.xticks(y_pos, objects)
    plt.ylabel('Chars No')
    plt.title('Language')
    #plt.xticks(index ('English', 'Other''))
    plt.show()

TotalChar= num_Enchars + num_Othchars
percent =  100 * float(num_Enchars)/float(TotalChar )

print ("Words  ", num_words)
print ("Number of English Chars  ", num_Enchars)
print ("Number of Non English Chars  ", num_Othchars)
print ("Total Chars  ", TotalChar)
print("Percentage",percent)

print(Histogram(x))
print(Histogram(y))
#print(Bar(x))
#print("Percentage",y)
```

Figure 4: Question 3



Important Notes

Submission

- Solutions have to be checked into the github repository. Use the directory name `groupname/assignment6/` in your group's repository.
- The name of the group and the names of all participating students must be listed on each submission.
- Solution format: all solutions as *one* PDF document. Programming code has to be submitted as Python code to the github repository. Upload *all* `.py` files of your program! Use `UTF-8` as the file encoding. *Other encodings will not be taken into account!*
- Check that your code compiles without errors.
- Make sure your code is formatted to be easy to read.
  - Make sure you code has consistent indentation.
  - Make sure you comment and document your code adequately in English.
  - Choose consistent and intuitive names for your identifiers.
- Do *not* use any accents, spaces or special characters in your filenames.

Acknowledgment

This latex template was created by Lukas Schmelzeisen for the tutorials of ”Web Information Retrieval”.

LaTeX

Currently the code can only be build using LuaLaTeX, so make sure you have that installed. If on Overleaf, there's an error, go to settings and change the LaTeXengine to `LuaLaTeX`.