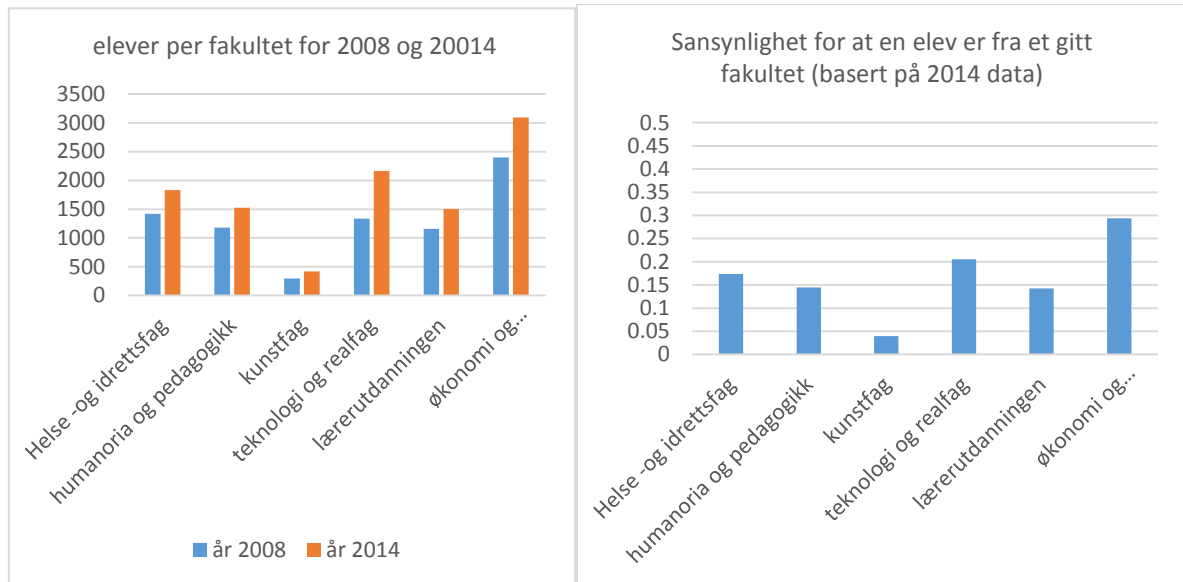


### 1.2.1

A.

«sansynlighet» er besvart med en presisjon på 2 desimaler.

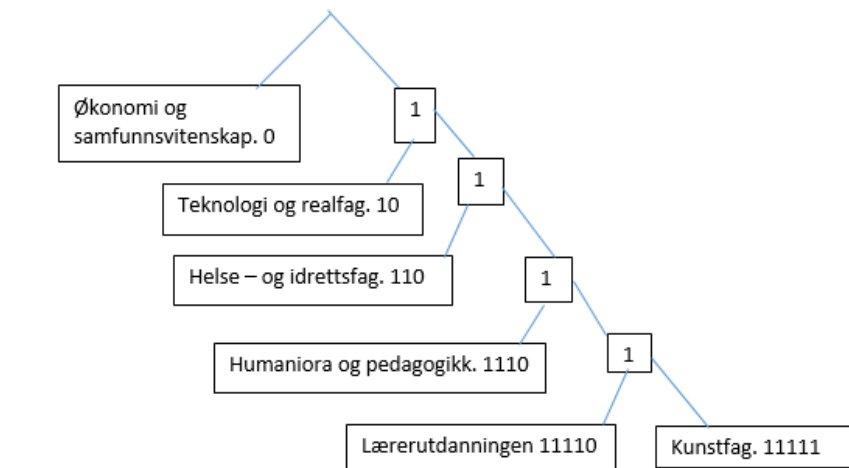
Pga begrensinger i Excel er det ikke mulig å få alt i en graf.



B. basert på sannsynlighetsfaktoren og hva vi har lært om komprimering vil du få minst informasjon dersom du får vite på at eleven tilhører fakultetet for «økonomi og samfunnsvitenskap».

C. Huffman treet er designet så den er «instantaneous» med opptil 5 bit kode, og om mindre enn 5 bit informasjon er gitt vil 0 være siste siffer i den binære koden for et fakultet.

(modellen og tabellen er formatert som bilde for å sikre formatering og kompatibilitet)



Fakultet	Kode	Kodelengde (i bits)
Økonomi og samfunnsvitenskap	0	1
Teknologi og realfag	10	2
Helse- og idrettsfag	110	3
Humaniora og pedagogikk	1110	4
Lærerutdanningen	11110	5
Kunsthøgskolen	11111	5

D.

Gjennomsnittslengden for meldingen med 100 tilfeldig utvalgte studenter blir 271,77 bits. Dette er kontrollregnet utfra mengden bits for hvert fakultet, og dens respektable sannsynlighet, summert sammen.

Utfra Shannons formel for utregning av entropi er entropien 2,41 for dette tilfellet. Altså gjennomsnittlig forventet bit per student er 2,41 bits, noe som betyr at optimalt vil man trenge 241 bits for å komprimere meldingen med 100 tilfeldig valgte studenter. I og med at koden vår er sub-optimal (dårlig komprimert) virker dette riktig.

E.

Algoritmen vil splitte opp det som skrives for hvert femte tegn (3 i den andre versjonen) og ved en while-løkke søke gjennom for å se om de passende kodene finnes i det som ble skrevet inn. Det vil deretter bli printet de passende fakultetene.

Selve koden er levert som en python fil i samme repository, og har navn «dekoder.py»