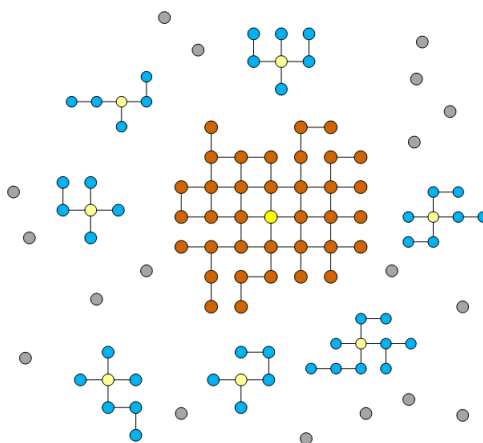


Uvod.

Podatki. Če je naloga zasnovana tako, da vključuje analizo izbranih podatkov, v tem razdelku opišeš, kakšni so ti podatki in navedeš nekaj osnovnih statističnih lastnosti teh podatkov. Slednje vključujejo velikost podatkov (na primer število primerov, število in vrsto atributov), delež manjkajočih podatkov, opis in porazdelitev vrednosti ciljnih spremenljivk, in podobno. Če si podatke pridobil sam, tu opišeš, na kakšen način, kje in kako.

Računanje razdalj. Tu opišeš, na kakšen način si rešil nalogo (tehnike in metode, ki si jih uporabil). Lahko vključiš tudi zanimiv del programske kode, ki si jo morda pri tem razvil ali pa v poročilo dodatno vključiš sliko, kot je na primer slika 1. Vse slike in tabele, ki jih vključiš v poročilo, morajo biti navedene v besedilu oziroma se moraš na njih sklicati.



Slika 1: Vsako sliko opremi s podnapisom, ki pove, kaj slika prikazuje.

Dendrogram V to poglavje lahko tudi vključiš kakšen metodološko zanimiv del kode. Primer vključitve kode oziroma implementirane funkcije v programskem jeziku Python je:

```
def fib(n):  
    if n == 0:  
        return 0  
    elif n == 1:  
        return 1  
    else:  
        return fib(n-1) + fib(n-2)
```

Izris te kode je lahko sicer tudi lepši, poskušaš lahko najti še primernejši način vključevanja kode v Pythonu oziroma v tvojem izbranem programskem jeziku v okolje L^AT_EX.

Skupine in njihove preferenčne izbire. V tem poglavju podaš rezultate s kratkim (eno-odstavčnim) komentarjem. Rezultate lahko prikažeš tudi v tabeli (primer je tabela 1).

Odstavke pri pisanju poročila v LaTeX-u ločiš tako, da pred novim odstavkom pustiš prazno vrstico. Tudi, če pišeš poročilo v kakšnem drugem urejevalniku, morajo odstavki biti vidno ločeni. To narediš z zamikanjem ali pa z dodatnim presledkom.

Tabela 1: Atributi in njihove zaloge vrednosti.

ime spremenljivke	definijsko območje	opis
cena	[0, 500]	cena izdelka v EUR
teža	[1, 1000]	teža izdelka v dag
kakovost	[slaba—srednja—dobra]	kakovost izdelka

Podajanje rezultati naj bo primerno strukturirano. Če ima naloga več podnalog, uporabi podpoglavja. Če bi želel poročati o rezultatih izčrpno in pri tem uporabiti vrsto tabel ali grafov, razmisli o varianti, kjer v tem poglavju prikažeš in komentiraš samo glavne rezultate, kakšne manj zanimive detajle pa vključi v prilogo (glej prilogi A in B).

Izjava o izdelavi domače naloge. Domačo nalogo in pripadajoče programe sem izdelal sam.

Priloge

Podrobni rezultati poskusov. Če je rezultatov v smislu tabel ali pa grafov v nalogi mnogo, predstavi v osnovnem besedilu samo glavne, podroben prikaz rezultatov pa lahko predstaviš v prilogi. V glavnem besedilu ne pozabi navesti, da so podrobni rezultati podani v prilogi.

Programska koda. Za domače naloge bo tipično potrebno kaj sprogramirati. Če ne bo od vas zahtevano, da kodo oddate posebej, to vključi v prilogo. Čisto za okus sem tu postavil nekaj kode, ki uporablja Orange (<http://www.biolab.si/orange>) in razvrščanje v skupine.

```
import random
import Orange

data_names = ["iris", "housing", "vehicle"]
data_sets = [Orange.data.Table(name) for name in data_names]

print "%10s_3s_3s_3s" % (" ", "Rnd", "Div", "HC")
for data, name in zip(data_sets, data_names):
    random.seed(42)
    km_random = Orange.clustering.kmeans.Clustering(data, centroids = 3)
    km_diversity = Orange.clustering.kmeans.Clustering(data, centroids = 3,
        initialization=Orange.clustering.kmeans.init_diversity)
    km_hc = Orange.clustering.kmeans.Clustering(data, centroids = 3,
        initialization=Orange.clustering.kmeans.init_hclustering(n=100))
    print "%10s_3d_3d_3d" % (name, km_random.iteration, \
        km_diversity.iteration, km_hc.iteration)
```