Harjoitukset 3

Harjoitustehtävissä käytetään seuraavia “UCI Machine Learning Repository” aineistoja:
• Letter Recognition dataset
• Wine dataset
Tehtäviä tehdessä kannattaa miettiä, mitä esiprosessointia (esim. scale() funktio) tekee vai tekeekö esiprosessointia ollenkaan. Tehtävissä voit valita käytettävän piirrejoukon vapaasti. Ristiinvalidointia muodostettaessa kannattaa käyttää “caret” paketin createFolds() funktiota.

1. Käy läpi K:n lähimmän naapurin menetelmän luentoesimerkit (R-koodit), jotka löytyvät Google Drivestä.

2. Tutkitaan “Wine” aineistoa.
a) Jaa “Wine” aineiston havainnot satunnaisesti opetusjoukkoon (60%), validointijoukkoon (20%) ja testijoukkoon (20%).
b) Tutki validointijoukon avulla, mikä k:n arvo on paras aineistolle. Käytä luokittelutarkkuutta (accuracy) arviointikriteerinä etsiessäsi parasta k:n arvoa.
c) Kun olet löytänyt parhaan k:n arvon, testaa, miten hyvin luokittelija luokittelee testijoukossa olevat havainnot. Määritä sekaannusmatriisi (table() funktio) ja luokittelutarkkuus sekaannusmatriisista.

3. Jatketaan “Wine” aineiston tutkimista.
a) Jaa “Wine” aineiston havainnot opetusjoukkoon (70%) ja testijoukkoon (30%) satunnaisesti.
b) Hae tällä kertaa optimaalinen k:n arvo soveltamalla 5-kertaista ristiinvalidointia opetusjoukkoon. Toista siis samaa 5-kertaista ristiinvalidointijakoa opetusjoukkoon käyttäen eri k:n arvoja. Hae jokaiselle testatulle k:n arvolle keskimääräinen tarkkuus.
c) Löydettyäsi parhaan k:n arvon, opeta luokittelija uudelleen käyttäen koko opetusaineistoa ja optimaalista k:n arvoa ja testaa, miten hyvin opetettu luokittelija luokittelee testiaineiston havainnot omiin luokkiinsa. Käytä luokittelutarkkuutta (määritetään sekaannusmatriisista esim. table() funktion avulla) arviointikriteerinä.

4. Jatketaan “Wine” aineiston analysointia. Oletetaan, että haluamme käyttää k:n lähimmän naapurin menetelmää luokitteluemenetelmänä ja haluamme käyttää vain k:n arvoa 3. Meillä on siis olemassa jotakin ennakkotietoa, jonka avulla on valittu k = 3.
Sovella koko aineistoon
(a) 10-kertaista ristiinvalidointia
(b) 5-kertaista ristiinvalidointia
(c) Leave-one-out-menetelmää.
Miten luokittelutulokset eroavat toisistaan? Viimeisessä tapauksessa kokeile myös, miten esiprosessoitu aineisto (scale() funktio) luokittuu leave-one-out-menetelmällä.

5. Tutustu lineaariseen ja neliölliseen diskriminanttianalyysiin (lda() ja qda() MASS paketissa). Nämä menetelmät eivät vaadi parametriarvojen optimointia. Näin ollen luokittelu onnistuu käyttäen esimerkiksi ristiinvalidointia koko aineistoon.
Jatketaan “Wine” aineiston analysointia. Sovella koko aineistoon
(a) 10-kertaista ristiinvalidointia
(b) 5-kertaista ristiinvalidointia
(c) Leave-one-out-menetelmää.
käyttäen lineaarista ja neliöllistä diskriminanttianalyysiä Miten luokittelutulokset eroavat toisistaan?

6. Sovella k:n lähimmän naapurin menetelmää, lineaarista ja neliöllistä diskriminanttianalyysiä “‘Letter Recognition” aineistoon.
Miten suunnittelet testausasetelman? Miten luokittelutulokset eroavat toisistaan?