doi:10.20312/dim.2015.02

# Az R szoftver alkalmazása az Adatbányászat tárgy oktatásában

**Pödör Zoltán**
NymE, SKK, Informatikai és Gazdasági Intézet
podor@inf.nyme.hu

ÖSSZEFOGLALÓ. Az adatbányászat egy tipikus informatikához köthető terület. A gazdaságinformatikus hallgatók számára különösen nagy jelentőséggel bír a gazdasági kötődés miatt. Bemutatjuk az általános elvárásokat az adatbányászati szoftverekkel szemben és konkrétan az R program alkalmazhatóságát az Adatbányászat oktatásában.

ABSTRACT. The data mining is a typical part of informatics science. It is especially important for the Business Information Technology students due to the economic specialization. We present the general expectations up to the data mining software and the application opportunities of R software in the education of Data mining.

## 1. Bevezetés

Az ember által generált adatmennyiség évente legalább megduplázódik. Rengeteg adatot gyűjtünk, gondoljunk csak az egyre jobban elterjedő szenzoros mérésekre. A hatalmas adathalmazok feldolgozása, a hasznos információk, összefüggések kinyerése nagy kihívást jelent. A hatalmas méretű adatbázisok, adathalmazok puszta adattemetők mindaddig, amíg fel nem dolgozzuk azokat. Ez a folyamat magában foglalja a hagyományos matematikai, statisztikai módszereket is. Azonban éppen az adathalmazok mérete teszi szükségessé olyan új megközelítések alkalmazását, ami lehetővé teszi ilyen esetben is a gyors és hatékony (sokszor online) feldolgozását. Ez több - akár hardver, akár szoftver oldalról - komoly informatikai kihívást jelent. Ezért fontos, hogy egy informatikus ismerje azokat a technikákat és eszközöket, melyek segítségével képesek vagyunk a Bigdata jellegű adatokból értékes információkat kinyerni és megjeleníteni. Bemutatjuk – mélyebb részletek ismertetése nélkül-, hogy a nyílt forrású R szoftverkörnyezet hogyan alkalmazható a tipikus adatbányászati feladatok megoldásában, a kapott eredmények vizuális reprezentációjában az NymE SKK gazdaságinformatikus hallgatóknak tartott Adatbányászat tárgy keretein belül.

## 2. Az adatbányászat

Az 1990-es évek elején a tárolókapacitások jelentős növekedése és ezzel párhuzamosan az árak csökkenése lehetővé tette, hogy gyakorlatilag az élet minden területén elterjedjenek a digitális eszközök és adatbázisok. Mindez azt eredményezte, hogy tömegesen keletkeztek nyers, feldolgozatlan adatok. Ezek feldolgozására akkor még csak a hagyományosnak tekinthető matematikai és statisztikai eljárások álltak rendelkezésre. Azonban ezek nem voltak alkalmasak az egyre növekvő méretű adathalmazokkal megbirkózni. Sem mennyiségben, sem a kinyert információk mélységében, finomságának tekintetében. Jól írja le az akkori helyzetet

John Naisbitt híres mondása: „Megfulladunk az információtól, miközben tudásra éhezünk.”. Az így megjelenő szükség keltette életre az adatbányászatnak nevezett tudományterületet.

Az adatbányászat [1], [2] az a folyamat, amellyel hasznos információk, mély, nemtriviális tudás nyerhető ki az adatbázisokból. Ehhez számtalan tudományterületet használunk fel, így az adatbányászat egy tipikus multidiszciplináris területnek tekinthető, mely a statisztikától kezdve, a mesterséges intelligencián keresztül az algoritmikus kérdésekig rengeteg területet ölel fel.

## 2.1. Adatbányászati szoftverek

Az alábbiakban összefoglaljuk azokat a kihívásokat [1], amiknek egy ilyen program meg kell, hogy feleljen. Ezek a problémák serkentői is voltak az adatbányászat fejlődésének.

- **Skálázhatóság**: képes legyen az egyre növekvő méretű adathalmazokat is hatékonyan kezelni.
- **Magas dimenzió**: alkalmas legyen a magas dimenzió számú adatbázisok kezelésére. Sok algoritmus esetében a számítási bonyolultság is gyorsan nő, ahogy a dimenziószám növekszik.
- **Heterogén és összetett adatok**: a hagyományos elemzési módszerek általában azonos típusú adathalmazokat képesek kezelni. Egyre fontosabb például a weben tárol félig strukturált szövegek kezelése, vagy a szociális hálókat reprezentáló, különböző formátumú adatokat tartalmazó gráfok feldolgozása.
- **Robusztusság**: hiányos, zajos adatokat is kezeljen, ez ne befolyásolja a működést.
- **Adatok tulajdonjoga, megosztása**: az utóbbi években különösen jellemző, hogy az elemzendő adatok fizikálisan sem egy helyen vannak eltárolva. Ez szükségessé tette az elosztott adatbányászati módszerek fejlesztését.

A piacon rengeteg szoftver áll rendelkezésre, röviden felsoroljuk, hogy mik azok a gyakorlati szempontok [1], amik egy jónak tekintett adatbányász szoftvert jellemeznek:

- **Algoritmusok**: az alkalmazott algoritmusok legyenek többszörösen megvalósítva (ahol erre lehetőség van), elvárás velük szemben a robusztusság és rugalmasság.
- **Vizualizációs lehetőségek**: elvárás az, hogy a kapott eredmények, összefüggések minél vizuálisabb módon is megjelenjenek. Ez az elvárás az ipari környezetben gyakran alkalmazott riportokkal szemben is. Ezek sokkal szemléletesebbé és áttekinthetőbbé teszik a táblázatokban tárolt számszerű eredményeket.
- **Be-, és kimeneti fájlformátumok**: az adott szoftver minél több fajta be-, illetve kimeneti formátumot tudjon alapértelmezésben kezelni. Az adatbányászatban tipikus, hogy adatok xls, sql, dbf, csv formában tárolunk, elvárjuk, hogy ezeket a rendszer képes legyen kezelni, mind a bemeneti, mind a kimeneti oldalon.
- **Felhasználóbarát**: az átlagos felhasználó olyan szoftvereket szeret használni, amelyek kezelése könnyen elsajátítható, megtanulható. Tipikusan ilyen pl. a RapidMiner által alkalmazott Drag and Drop technika. Ugyanakkor fontos megemlíteni, hogy ez bizonyos korlátokat is hordoz magában.

Az elérhető szoftverek között vannak fizetősök, mint például az IBM SPSS, SAS, RapidMiner, Statistica Dataminer, Oracle, Microsoft Analysis Services és szinte megszámlálhatatlan nyílt kódú szoftvert lehet összeszedni. A teljesség igénye nélkül néhány: Weka, R, Orange, SCaViS, Knime, Octave Adam. Jelen munkának nem célja kitérni ezekre a szoftverekre, azok előnyeire és hátrányaira. Azonban az interneten fellelhető több olyan oldal [3], ahol ezeket rangsorolják különböző szempontok figyelembe vétele mellett, mint például népszerűség, hatékonyság, jóság. Ezek a statisztikák természetesen nem tekinthetőek reprezentatívnak, sőt sokszor erősen szubjektív elemeket is tartalmaznak, azonban mégiscsak mutatnak egyfajta képet ezekről a szoftverekről. Az R szoftver mind a két felsorolás elején

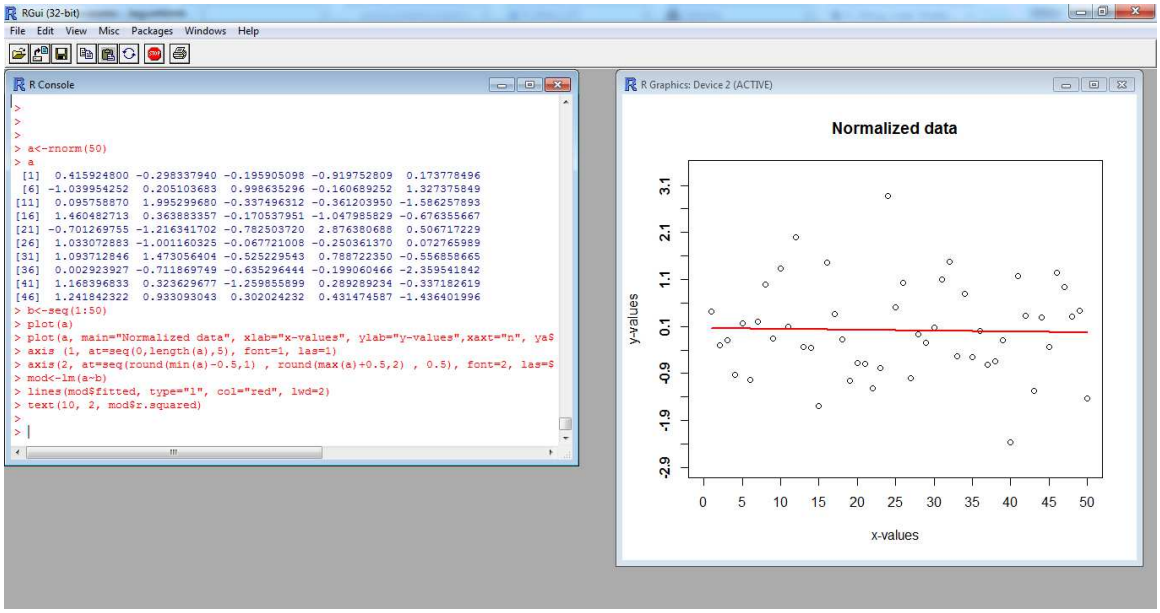
helyezkedik el és figyelembe véve, hogy a felhasználók szempontjából kevéssé kedvelt szkript nyelvről van szó, ez meglehetősen kedvező pozíció.

# 3. R program

## 3.1. A program rövid bemutatása

Az R egy olyan nyílt forráskódú, szkript alapú nyelv [5], amely különösen alkalmas matematikai, statisztikai és adatelemzési, adatbányászati számítások megvalósítására és az eredmények grafikus megjelenítésére. A program bárki által szabadon fejleszthető, így számtalan publikált csomag és függvény áll rendelkezésére a felhasználóknak. A program egyszerű használatához nem kell feltétlenül informatikusnak lenni, azonban valódi mélységei és lehetőségei programozói ismeretek birtokában tárulnak fel.

Nagyon fejlett és bőséges eszköztárral rendelkezik, és nem csupán matematikai, statisztikai módszerek terén, bár minden olyan feladat megoldható ebben a környezetben, ami például a Mathlab vagy a Maple programokkal. Elérhetünk például hagyományos, relációs vagy NOSQL alapú adatbázisokat és minden fontos adatbányászati algoritmus is implementálásra került. Támogatott benne a hálózati kommunikáció, de lehet akár grafikus felhasználói felületeket is készíteni. Egyik legnagyobb erőssége éppen grafikai képességeiben rejlik. Lehetőség van továbbá az R nyelv webes környezetben való működtetésére is az rApache [4] megoldással.




**1. ábra. Az R program**

Az R-nyelv egy interpretált szkript nyelv, a programkódokat nem fordítjuk bináris állománnyá a futtatáshoz, hanem az R-parancsértelmező értelmezi azokat. Jelen munkának nem célja az R nyelv alapjainak ismertetése, sokkal inkább, hogy néhány jellegzetes adatbányászati feladaton keresztül bemutassuk az R nyelv alkalmazhatóságát. A továbbiakban kiválasztottunk néhány tipikus, az Adatbányászat tárgy keretein belül oktatott témát, amiknek segítségével illusztráljuk az R nyelv működését, programozhatósági és grafikus képességeit, lehetőségeit. A feladatok illusztrálására a szabadon hozzáférhető iris adathalmaz használtuk fel.

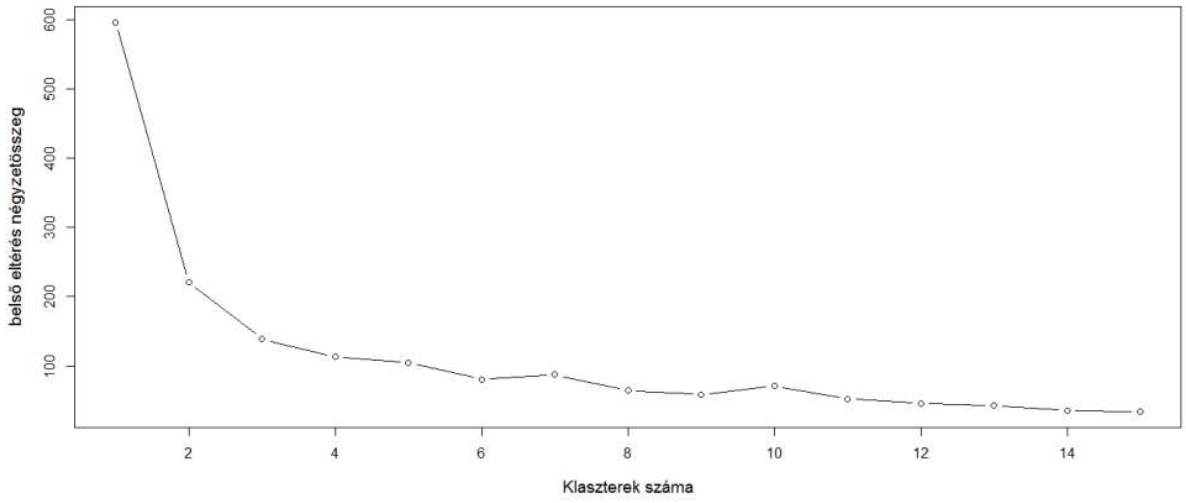
## 3.2. A program alkalmazhatósága az adatbányászatban

Két tipikus adatbányászati feladatot: klaszterezés, osztályozás, illetve egy a hagyományos statisztikában is gyakran alkalmazott témát, a regressziót választottuk ki [1], [2]. Terjedelmi okok miatt nem célunk a fenti eljárások mély elméleti ismertetése, hanem röviden bemutatjuk az alapproblémát és megoldási lehetőségeket az R környezetben. Mindegyik feladat esetében megkerülhetetlen lépés az adatok megfelelő előkészítése, ami magában foglalja a hiányos, zajos adatok kezelését, kiugró adatok szűrését, illetve az adott feladathoz kötődő egyéb műveleteket.

### *Klaszterezés*

Szokás a klaszterezést felügyelet nélküli tanulásnak is nevezni, melynek célja, hogy a vizsgált elemeket olyan csoportokba osszuk, ahol az egyes csoportok között maximális, míg a csoportokon belül minimális a távolság/hasonlóság. A klaszterezés célja, olyan csoportosítások kialakítása, melyek triviálisan nem látszanak az alapadatokból és nem használunk előfeltevéseket a csoportok kialakítása során.

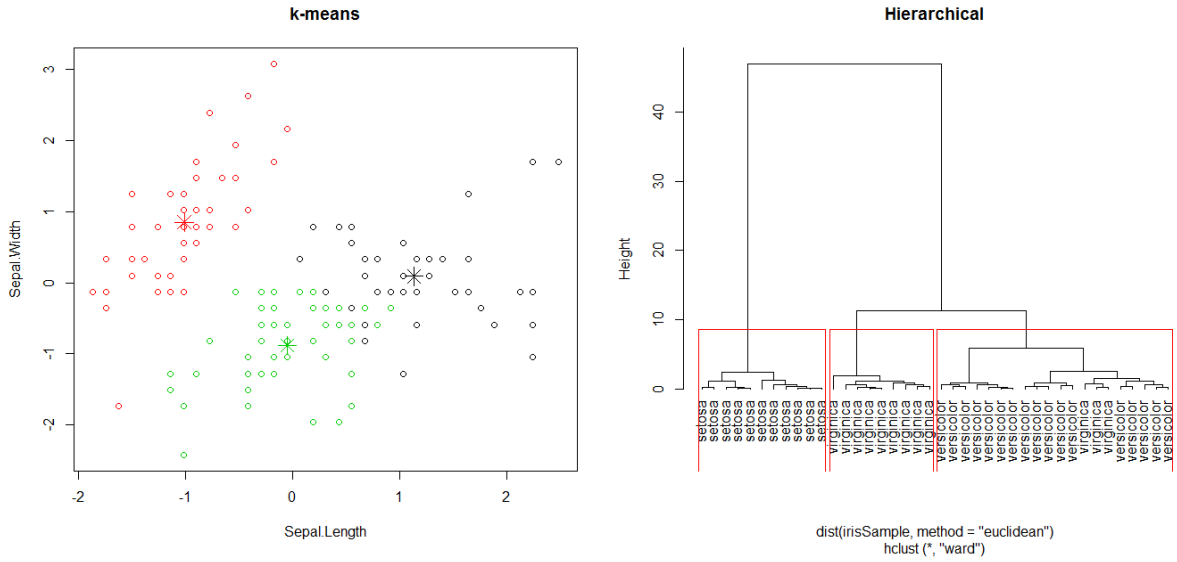
A klaszterezési feladatban nehézséget jelent az alkalmazandó távolság fogalom kiválasztása, a különböző típusú adatok együttes kezelése, és a különböző alakú klaszterek felismerése. Több, különböző klaszterező módszer létezik, mint a particionáló, hierarchikus, sűrűség-, rács-, modell-alapú módszerek stb. Mi a két legáltalánosabban elterjedt megközelítést alkalmazzuk, a felosztó és a hierarchikus megközelítést. Előbbi esetben fontos kérdés a klaszterek számának meghatározása, ami a módszer kötelező bemenő paramétere.




**2. ábra. Könyökpont módszer**

```
wss <- (nrow(mydata)-1)*sum(apply(mydata,2,var))
for (i in 2:15) wss[i] <- sum(kmeans(mydata, centers=i)$withinss)
plot(1:15, wss, type="b", xlab="Klaszterek száma", ylab="belső eltérés négyzetösszeg")
```

A könyökpont módszer segítségével vizuálisan becsülhetővé válik az optimális klaszterszám, amit így már felhasználhatunk, mint a folyamat bemenő paraméterét. A hierarchikus klaszterezés esetében utólagosan definiálható az ideális klaszterszám, amit vizuálisan a piros téglalapok jelölnek (3. ábra). Az eredmények természetesen táblázatos formában is lekérhetőek, ahol minden sor mellett látható, hogy melyik klaszterbe tartozik.



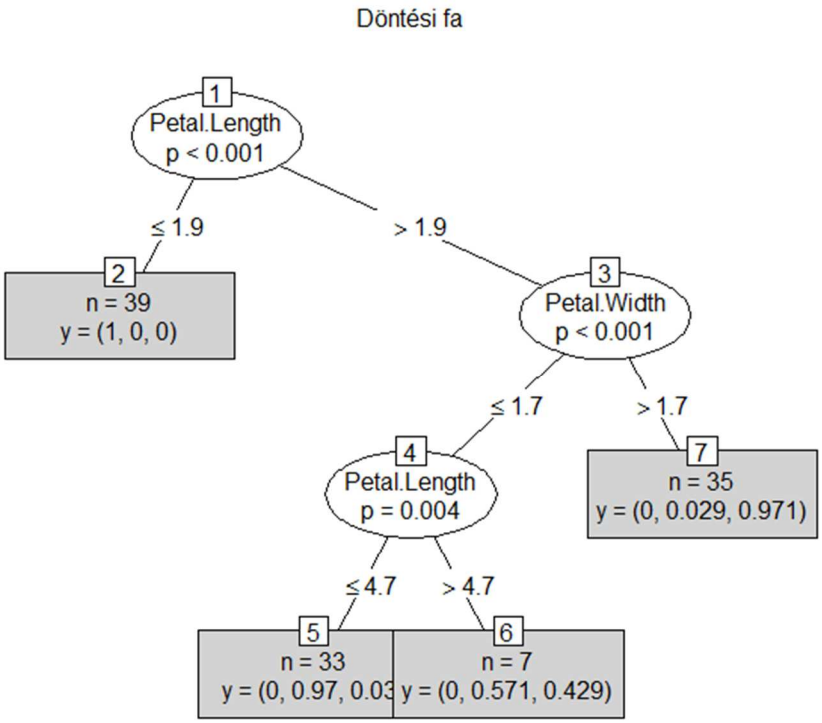

**3. ábra. K-means és hierarchikus klaszterezés**

```
idx <- sample(1:dim(iris)[1], 40)
irisSample <- iris[idx,]
irisSample$Species <- NULL
kc <- kmeans(mydata, 3)
par(mfrow=c(1,2))
plot(mydata[,c("Sepal.Length", "Sepal.Width")], col = kc$cluster, main="k-means")
points(kc$centers[,c("Sepal.Length", "Sepal.Width")], col = 1:3, pch = 8, cex=2)
hc <- hclust(dist(iris, method="euclidean"), method="ward")
plot(hc, hang = -1, labels=iris$Species[idx], main="Hierarchical")
rect.hclust(hc, k=3, border="red")
```

## *Osztályozás*

A klaszterezéssel ellentétben ezt felügyelt tanulásnak nevezzük, mert előre ismertek az osztálycímkék, ami alapján a csoportok, osztályok kialakításra kerülnek. A rendelkezésre álló adathalmazt felosztjuk egy tanuló és egy validáló halmazra. A tanuló halmazon hozzuk létre a modellt, melynek jóságát a validáló halmazon ellenőrizzük. Az osztályozási feladat célja a kialakított és validált modell alapján előrejelzések készítése. Mint például előzetes hitelbírálat, banki csalások detektálása, weben megjelenő oldalak automatizált besorolása, stb..

A feladat elején definiálnunk kell a függő és magyarázó változókat, utóbbiak segítségével fogjuk a függő paraméter értékei alapján osztályozni adatainkat. Számtalan osztályozási technika létezik, mint a döntési-fa, Bayes-alapú, legközelebbi szomszéd osztályozók, neurális hálók, tartóvektor gépek (SVM). A legismertebb osztályozási módszerre, a döntési-fa alapú osztályozásra mutatunk példát.

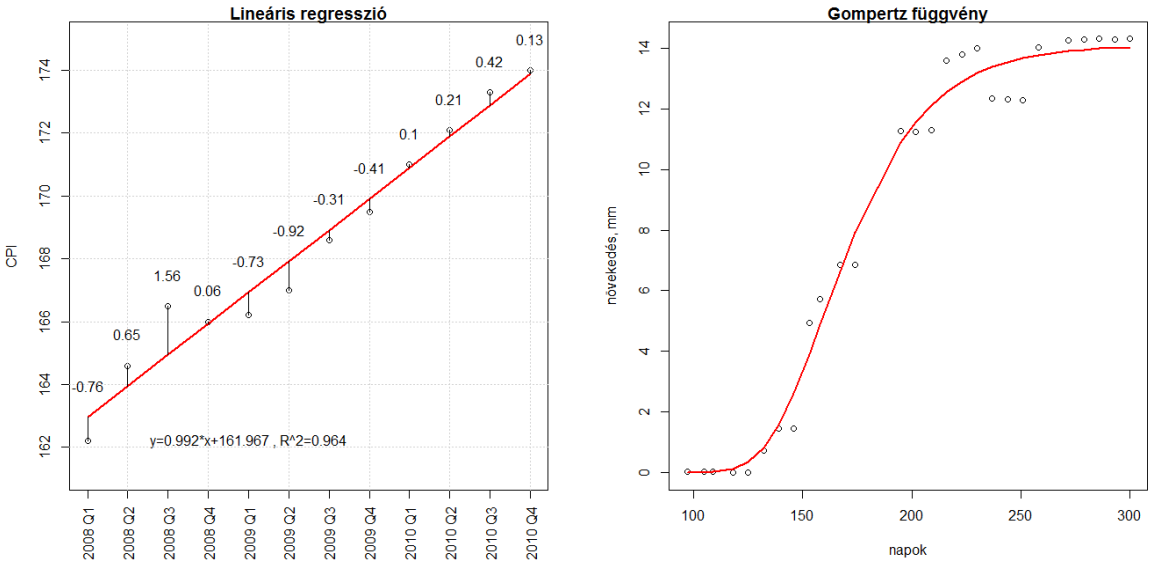



**4. ábra. Osztályozás – döntési fa**

```
tesztdata<-subset(mydata, row.names(mydata) %in% teszt)
tanulodata<-subset(mydata, row.names(mydata) %in% tanulo)
dectree1<-ctree(Species~Sepal.Length+Sepal.Width+Petal.Length+Petal.Width, data=tanulodata)
plot(dectree1, type="simple", main="Döntési fa")
```

## *Regresszió*

Az egy-, és többváltozós, lineáris-, és nemlineáris regresszió a statisztika egy jól ismert és gyakran alkalmazott módszertana. Segítségével függvény jellegű kapcsolat definiálható a függő és független paraméterek között, ami alkalmas a változók közötti kapcsolatok jellemzésére, előrejelzések készítésére a függő paraméter vonatkozásában. Gyakran alkalmazott technika idősorok esetében a trend, tendencia meghatározásában, adatelőkészítő folyamatban (zajos, hiányos adatok kezelése). A nemlineáris regresszió, már egyváltozós esetben is komoly matematikai apparátus alkalmazását követeli meg (pl. nemlineáris egyenletrendszer megoldása), ami többváltozós esetben még tovább bonyolódik, hiszen sokszor már az alkalmazandó függvény meghatározása sem könnyű feladat.




**5. ábra. Regressziós görbe illesztése**

Két példán keresztül bemutatjuk, hogy mennyire könnyű az R nyelv alkalmazásával elvégezni az illesztéseket, előállítani a modell paramétereit és felhasználni azt az előrejelzésekben. Ugyanakkor mindezt kiegészíti az a grafikus háttér, ami rendkívül szemléletessé teszi a kapott eredményeket, segítve azok könnyebb gyakorlati értelmezését.

A vizuális megjelenítéssel együtt könnyen generálhatóak az illesztett görbét jellemző egyéb paraméterek: regressziós egyenlet, determinációs együttható, reziduálisok, illesztett értékek, előrejelzett értékek.

## 4. Összefoglaló

Az adatbányászat alapjainak ismerete elengedhetetlen egy informatikus végzettségű embernek. Akár ipari, akár tudományos környezetben olyan mennyiségű adat generálódik, aminek a hatékony feldolgozása ma már elképzelhetetlen a hagyományos statisztikai eszközök mellett adatbányászati megoldások alkalmazása nélkül. Ezek tipikusan olyan megoldások, amelyek ötvözik az informatikára jellemző algoritmikus, és a statisztikára jellemző matematikus gondolkodást.

Ahogy bemutattuk, számtalan fizetős és még több nyílt megoldás létezik adatbányászati feladatok megvalósítására. Ezek közül az R szoftvert mutattuk be röviden, inkább csak felvillantva lehetőségeit ezen a területen. Ehhez három, az adatbányászatra tipikusan jellemző feladatot választottunk ki: klaszterezés, osztályozás és regresszió. Az iris adathalmazon keresztül mutattuk be a lehetőségeket.

## Irodalomjegyzék

[1] **Bodon F**.:, Adatbányászati algoritmusok, jegyzet (2010).
[2] **Han, J., Camber, M.,** Data Mining, Concepts and Techniques – second edition, *Morgan Kaufmann Publishers* (2006) pp. 772.
[3] http://www.predictiveanalyticstoday.com/
[4] http://rapache.net/
[5] https://www.r-project.org/