

EXCEL VBA ŰRLAP KIALAKÍTÁSA EGYMINTÁS T-PRÓBA KIÉRTÉKELÉSÉHEZ

Hampel György

Absztrakt: A Microsoft Excel program Visual Basic for Applications szolgáltatásával kialakítottunk egy felhasználóbarát felületet egymintás t-próba elvégzésére. Így akár az Excel táblázatkezelő használatának ismerete nélkül is lehetőség van az egyik leggyakrabban megjelenő hipotézisvizsgálat, a t-próba elvégzésére. A használathoz elegendő megnyitni az elkészített alkalmazást, s a megjelenő ablakban a szükséges adatok és opciók beállítása után megkapjuk az eredményeket. A program vezérlőkkel biztosítja a könnyű kezelhetőséget, míg a működést eseményvezérelt programozási technikával alakítottuk ki. Az eredményeket megkaphatjuk a betáplált minta adatsor alapján is, de a minta főbb jellemzői, az elemszám, az átlag és a szórás értékéből is. A vizsgálandó hipotézis megadásakor tetszőleges reláció alkalmazható, s ettől függően kétszélű vagy egyszélű eset egyikeként, automatikusan kapjuk meg az eredményeket. Az alkalmazás használatával matematikai statisztikai ismeretek és statisztikai programcsomag nélkül végezhetjük el az egymintás t-próbát.

Abstract: Using Visual Basic for Applications in Microsoft Excel, we designed a user-friendly interface for performing a one-sample t-test. Thus, even without knowledge of the use of the Excel spreadsheet, it is possible to perform one of the most common hypothesis tests, the t-test. All you have to do is open the completed application and in the window that appears, you will get the results after setting the required data and options. The program provides easy operation with controls, while the operation is designed with event-driven programming techniques. The results can be obtained from the data set of the fed sample, but also from the main characteristics of the sample, the number of items, the mean and the standard deviation. When specifying the hypothesis to be examined, any relation can be used, and depending on this, we obtain the results automatically as one of the double-edged or single-edged cases. Using the application, we can perform the one-sample t-test without mathematical statistical knowledge and statistical software package.

Kulcsszavak: matematikai statisztika, egymintás t-próba, Excel VBA, programozás

Keywords: mathematical statistics, one-sample t-test, Excel VBA, programming

1. Bevezetés

Az informatikai eszközök térnyerésével folyamatosan több adat jelenik meg és tárolódik. Ezek feldolgozása, értelmezése és hasznosítása napjaink egyik nagy kihívása. Több publikáció jelent meg olyan alkalmazások fejlesztéséről, mely adatbázis szolgáltatásokat biztosít Access környezetben (Fabulya, 2018a; Fabulya, 2018b). Az adatok rendszerezett tárolása teszi lehetővé, hogy döntéseink adatokkal kellően alátámasztottak lehessenek. Sokszor matematikai statisztikai módszerek nélkülözhetetlenek az adatok kiértékeléséhez, melyhez kellő szakértelem szükséges még statisztikai programcsomag használata mellett is (Móri, 2011).

Célul tűztük ki, hogy a statisztikai kiértékelések elvégzését segítő alkalmazásokat fejlesszünk ki Microsoft Excel táblázatkezelővel, kihasználva annak programozhatósági lehetőségét a Visual Basic for Applications (VBA) szolgáltatást. Ennek keretében készítjük el az egyik legtöbbször alkalmazott hipotézisvizsgálat, az egymintás t-próba elvégzését megvalósító programunkat. Fontos szempont munkánk során, hogy olyan felületet alakítsunk ki a program használatához, mely nem igényel speciális jártasságot sem a számítógép kezelésében, sem a matematikai statisztika

terén. Az Excel VBA használatával könnyen kialakíthatunk matematikai számításokat segítő alkalmazásokat (Fabulya, 2019), de akár dokumentumok generálását is VBA program végezheti (Fabulya, 2020).

A továbbiakban bemutatjuk a kutatás alapját biztosító egymintás t-próba elméleti ismereteit, valamint a VBA alkalmazott eszközeit, technikáit. Ezt követően részletesen ismertetésre kerül az elkészült program minden fontosabb eleme.

2. Anyag és módszer

2.1. Az egymintás t-próba

Az egymintás t-próbával egy normális eloszlású statisztikai sokaság várhatóértékére (m) vonatkozó hipotézist tesztelhetjük egy, a sokaságból származó minta alapján. A t-próba végrehajtásakor elvégzendő számítás elméleti háttere azon alapul, hogy feltételezve a sokaság várhatóértéket a hipotézisben megjelenő értékűnek (m_0), ekkor a minta átlaga (\bar{x}) nagy valószínűséggel e feltételezett érték közelében lesz. Így a döntéskor csak kis valószínűséggel követünk el hibát akkor, amikor a feltételezett várhatóértéket nem tartjuk elfogadhatónak a minta átlagától való jelentős, szignifikáns eltérése miatt, pedig a várhatóérték ennek ellenére a feltételezett értékű. Vagyis elsőfajú hibát csak tetszőlegesen kis valószínűséggel követhetünk el, mert ezt az elsőfajú hibavalószínűséget (ε) a próba végrehajtása elején magunk választhatjuk meg (Michaletzky–Mogyoródi, 1995).

A próba végrehajtásakor először formálisan egy nullhipotézist (H_0) és egy alternatív hipotézist (H_1) állítunk fel. A nullhipotézis mindig a várhatóérték (m) egyenlőségét tartalmazza egy értékkel (m_0), míg az alternatív hipotézis az egyenlőségtől eltérést három formában jelezheti, alkalmazkodva a vizsgálandó hipotézishez az 1. táblázat szerint.

1. táblázat: Null és alternatív hipotézis

Vizsgálandó hipotézis	H_0	H_1
$m = m_0$	$m = m_0$	$m \neq m_0$
$m \neq m_0$	$m = m_0$	$m \neq m_0$
$m > m_0$	$m = m_0$	$m > m_0$
$m < m_0$	$m = m_0$	$m < m_0$
$m \geq m_0$	$m = m_0$	$m < m_0$
$m \leq m_0$	$m = m_0$	$m > m_0$

Forrás: a szerző saját szerkesztése.

A mintaelemek (x_i) ismeretében kiszámítjuk az átlagot és a minta szórását az (1) és (2) képlet szerint.

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (1)$$

$$s = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (2)$$

ahol:

\bar{x} = a minta átlaga

x_i = az i-edik mintaelem értéke

n = a minta elemszáma

s = a minta szórása

A hipotézisvizsgálat végén a döntést az alternatív hipotézis elfogadásáról, vagy elutasításáról két technikával végezhetjük, a kritikus tartománnyal, illetve a próba szignifikanciaszintjével. Ehhez elsőként ki kell számítani az egymintás t-próba statisztikai függvényének az értékét a (3) képlettel.

$$t = \frac{\bar{x} - m_0}{s} \cdot \sqrt{n} \quad (3)$$

ahol:

t = a statisztikai függvény értéke

\bar{x} = a minta átlaga

m_0 = a statisztikai sokaság feltételezett várhatóértéke

n = a minta elemszáma

s = a minta szórása

Mivel a statisztikai függvény értéke $(n - 1)$ szabadságfokú t-eloszlást követ, ezért a kritikus tartomány határát, illetve a próba szignifikanciaszintjét a t-eloszlás eloszlásfüggvénye alapján kapjuk meg. A kritikus tartomány azon értékek intervalluma, melybe, ha a statisztikai függvény értéke tartozik, akkor az alternatív hipotézist fogadjuk el. A próba szignifikanciaszintje az a legnagyobb elsőfajú hibavalószínűség, melynél még a nullhipotézist fogadjuk el. Ezért, ha az elsőfajú hibavalószínűség nagyobb a próba szignifikanciaszintjénél, akkor az alternatív hipotézist fogadjuk el (Obádovics, 2020).

A (4) képlet mutatja az eloszlásfüggvény és a valószínűség közötti kapcsolatot. Együttal a képlet tekinthető a kvantilis definíciójának.

$$P(t < t_p) = F(t_p) = p \quad (4)$$

ahol:

t = egy t-eloszlású valószínűségi változó értéke

t_p = a t-eloszlás p valószínűséghez tartozó kvantilise

F = a t-eloszlás eloszlásfüggvénye

Amikor a döntést a kritikus tartomány alapján végezzük, akkor a statisztikai függvény kiszámított értékét ahhoz a kvantilishoz viszonyítjuk, mely a kritikus tartomány határa. Ezért ez a kvantilis függ az elsőfajú hibavalószínűségtől és az

alternatív hipotézistől. A próba szignifikanciaszintjének kiszámításának alapja a statisztikai függvény értéke, de az alternatív hipotézistől is függ. A döntésnél alkalmazott feltételt és a szignifikanciaszint kiszámítását a 2. táblázat tartalmazza.

2. táblázat: A döntés feltétele és a szignifikanciaszint kiszámítása

H_1	H_1 elfogadási feltétele kritikus tartománnyal	Szignifikanciaszint (p)
$m \neq m_0$	$ t > t_{1-\frac{\varepsilon}{2}}$	$p = 2 \cdot (1 - F(t))$
$m < m_0$	$t < t_\varepsilon$	$p = F(t)$
$m > m_0$	$t > t_{1-\varepsilon}$	$p = 1 - F(t)$

Forrás: a szerző saját szerkesztése.

2.2. Az Excel VBA

A Microsoft Excel táblázatkezelő program használatát, a szükséges számítások elvégzését, az eredmények megjelenítését egyedivé, felhasználóbaráttá és automatizálhatóvá tehetjük a Visual Basic for Applications (VBA) használatával (Matteson, 1995). A számítások kiinduló adatainak megadása és az eredmények megjelenítése nem csak a munkalapok celláival valósítható meg. Különböző vezérlőket alkalmazhatunk, melyeket elhelyezhetünk a munkalapokon, vagy űrlapokon (UserForm). Így magunk tervezhetjük meg a felhasználói felületet, melyen az adatbevitel, az eredmények megjelenítése és a számításokhoz szükséges opciók, beállítások az arra alkalmas vezérlőkkel megvalósíthatók. Eseményvezérelt programozási technika biztosítja, hogy a használat során a felhasználó tevékenységére reagáló viselkedést érzünk el. Ilyen felprogramozható esemény lehet klikkelés az egérrel egy parancsgombon, egy opció megváltoztatása, az adatbevitel végén az Enter leütése stb. Így például rögtön ellenőrizhetjük az adat helyességét, hiba esetén figyelmeztető üzenetet jeleníthetünk meg, blokkolhatunk tevékenységeket, amíg hibát detektálunk. A számítások végrehajtását programunk segítségével automatizálhatjuk, eredményeket írhat a programunk a munkalapokra, de a munkalapokon számított értékeket is feldolgozhatjuk (Kovalcsik, 2005).

Gyakrabban használt vezérlők:

- Űrlap (UserForm): A kezelőfelületet ablakként biztosítja, melyen további vezérlők helyezhetők el.
- Beviteli mező (TextBox): Elsődlegesen adat begépelésre kialakított vezérlő, de eredmény is megjeleníthető vele.
- Parancsgomb (CommandButton): Ráklizelve kezdeményezhető valamilyen tevékenység végrehajtása.
- Jelölőnégyzet (CheckBox): Felirattal rendelkező vezérlő, melynek be- és kikapcsolt állapotát klikkeléssel egyszerűen beállíthatjuk.
- Választókapcsoló (OptionButton): Csoportosan alkalmazzuk, közülük csak egy lehet bekapcsolt állapotú.
- Listapanel (ListBox): Adatok listájának megjelenítését teszi lehetővé, de kijelölést is alkalmazhatunk a listában.

- Legördülő lista (ComboBox): Legtöbbször arra használjuk, hogy kiválasszunk a listából egy elemet. Helytakarékos, mert csak a kiválasztott értéket mutatja, ha már választottunk a legördíthető listából.
- Felirat (Label): Feliratok megjelenítésére alkalmas. Így eredmények is kijelvezhetők vele, mert a programunk megváltoztathatja a vezérlő által tárolt szöveget.
- Több oldal (MultiPage): Akkor célszerű alkalmazni, amikor sok vezérlő szükséges, ami miatt túl nagy lenne az űrlap. Lapozhatóan több oldalt biztosít a vezérlők elhelyezésére.
- Keret (Frame): Vezérlők csoportosítására használhatjuk, hogy rendszerezett, jobban áttekinthető űrlapot alakíthassunk ki.

A vezérlők tulajdonságokkal és metódusokkal rendelkező objektumok. A tulajdonságaik egy-egy értéket tárolnak, melyek meghatározzák a vezérlő megjelenését, viselkedését. A tulajdonságokat legtöbbször előre, a program tervezési fázisában beállítjuk, de futási időben a programunk is módosíthatja őket. A vezérlők metódusai programok, melyeket futtatva valamilyen tevékenységet hajt végre a vezérlőnk (Zimmerman, 1996).

3. Eredmények és értékelésük

A felhasználóbarát megjelenés és használat alapos tervezést igényel már az űrlap kialakításakor is. Az alábbi szempontokat tűztük ki a megvalósuló program tulajdonságaiként:

- A felhasználó minden szükséges tevékenységet egy űrlapon tudjon elvégezni. Az űrlap automatikusan jelenjen meg az Excel fájl megnyitásakor.
- Külön lap biztosítsa az adatbevitelt és az eredmények megjelenítését. Ezt MultiPage vezérlővel oldhatjuk meg. Így az eredmények kiszámítása csak akkor történik meg, amikor átlapozunk az adatbeviteli lapról az eredmények lapjára. Tehát parancsgomb sem szükséges a számítások végrehajtásához.
- Minden lehetséges adatból elvégezhető legyen a kiértékelés. A felhasználó tudja kiválasztani egy jelölőnégyzettel, hogy a minta adatsorán vagy a minta jellemzőin (elemszám, átlag, szórás) alapuljon a számítás.
- Bármilyen relációval megadható hipotézis kiértékelhető legyen. A lehetséges hipotézisek egyikének kiválasztását egy legördülő listából lehessen kiválasztani.
- A számítások egy Excel munkalapon, a felhasználótól rejtetten történjenek. Így nem kell a számításokat a programunknak tartalmaznia, csak az eredményeket kell átvenni a munkalapról, hogy az űrlapon jelenjenek meg.

3.1. Az adatok megadásának felhasználói felülete

Az egymintás t-próba kiértékelését végző program indításakor megjelenő felhasználói felületet az *1. ábrán* láthatjuk.

1. ábra: A felhasználói felület

Forrás: a szerző saját szerkesztése.

Kezdetben az „Adatok” lapot használjuk. Az űrlap automatikus megjelenéséhez az Excel fájl megnyitásának eseményét kellett felprogramozni, mely az alábbi VBA kóddal történt.

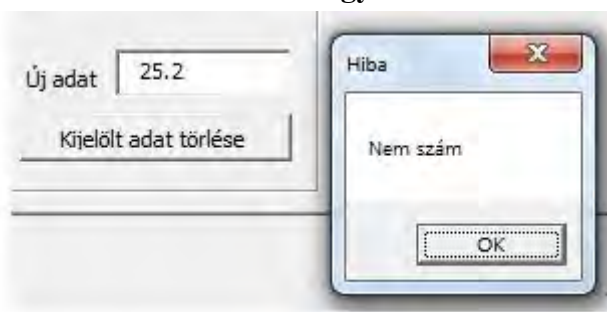
```
Private Sub Workbook_Open()  
    UserForm1.Show  
End Sub
```

Az esemény (`Workbook_Open`) bekövetkezésekor egyetlen utasítás hajtódik végre a `UserForm1` nevű űrlap vezérlő `Show` metódusa, mely megjeleníti az űrlapot. A jobb oldalon látható, hogy egy legördülő listából választhatjuk ki a hipotézis típusát, míg alatta egy-egy beviteli mező már alapértelmezett értékeket tartalmaz az elsőfajú hibavalószínűség és a feltételezett várhatóérték adataiként. Ezek természetesen módosíthatók.

3.1.1. Ismert adatsor megadása

Az űrlap bal oldalán egy jelölő négyzettel állíthatjuk be, hogy ismert adatsor esetén ez megadható legyen a „Minta adatai” feliratú keretben. Az új adat megadása beviteli mezővel történik. Felhasználóbarát megoldás, hogy az új adat begépelésének végén, az Enter leütésekor ellenőrzi a programunk az adat érvényességét. Ha nem tekinthető számnak az új adat, akkor hibaüzenetet kapunk (2. ábra), majd el kell végeznünk a javítást, hogy az érvényes adat bekerülhessen a felette megjelenő, kezdetben üres listába.

2. ábra: Hibaüzenet egy adat esetén



Forrás: a szerző saját szerkesztése.

A hiba ellenőrzését a beviteli mező billentyű leütés (TextBox1_KeyDown) eseményéhez rendeltük. A program VBA kódja az alábbi:

```
Private Sub TextBox1_KeyDown(ByVal KeyCode_
    As MSForms.ReturnInteger, ByVal Shift As Integer)
    If KeyCode = 13 Then
        ent = True
        If IsNumeric(TextBox1.Text) Then
            ListBox1.AddItem TextBox1.Text
            TextBox1.Text = ""
            TextBox1.SetFocus
        Else
            MsgBox "Nem szám", , "Hiba"
        End If
    End If
End Sub
```

Csak *Enter* leütésekor (KeyCode = 13) kell ellenőrizni, hogy a beviteli mezőben lévő adat (TextBox1.Text) számnak tekinthető-e (IsNumeric). Ha igen, akkor az adat bekerül a listába (ListBox1.AddItem), majd kiürítjük a beviteli mezőt, hogy az újabb adat begépelését előkészítsük. Ha hibás az adat, akkor megjelenítjük a hibaüzenetet.

Az ent = True értékadás jelzi, hogy az Enter billentyűt ütöttük le. Ez szintén a felhasználóbarát működéshez kell, hogy meg tudjuk akadályozni, hogy egy másik vezérlőre térjen át a programunk. Ez bekövetkezne, ha programunk nem

gondoskodna erről. Az alábbi programrész biztosítja, hogy folyamatosan gépelhessük be az újabb adatokat anélkül, hogy vissza kellene kattelnünk a beviteli mezőbe:

```
Private Sub TextBox1_Exit(ByVal Cancel_
                        As MSForms.ReturnBoolean)
    If ent Then
        Cancel = True
        ent = False
    End If
End Sub
```

Tehát a `TextBox1_Exit` eseményét programoztuk fel úgy, hogy ha az Enter leütése miatt lépnénk ki a beviteli mezőből, akkor a kilépést blokkoljuk (`Cancel = True`).

3.1.2. Adatsor jellemzőinek megadása

Ha a minta adatsora nem ismert, akkor a minta három jellemzőjének megadása szükséges az egymintás t-próba kiértékeléséhez. Kikapcsolva az „Adatsor ismert” jelölő négyzetet, az űrlapon a minta jellemzőit tudjuk begépelni egy-egy beviteli mezőben. Ez látható a 3. ábrán. Ekkor a „Minta jellemzői” feliratú keret jelenik meg a benne lévő vezérlőkkel, leváltva a „Minta adatai” keretünket. Ezt az alábbi programkód eredményezi:

```
Private Sub CheckBox1_Change()
    Call kinezet
End Sub
Public Sub kinezet()
    If CheckBox1 Then
        Frame1.Visible = True
        Frame2.Visible = False
    Else
        Frame1.Visible = False
        Frame2.Visible = True
    End If
End Sub
```

Tehát a jelölő négyzet megváltozásakor (`CheckBox1_Change`) meghívjuk a `kinezet` nevű eljárást végrehajtását, melyben a keret vezérlők láthatóságának tulajdonságát állítjuk át (`Frame1.Visible` és `Frame2.Visible`). Azért került mindez egy kinézetet beállító eljárásba, mert erre a változásra nem csak a jelölő négyzet megváltozásakor van szükség, hanem a kezdeti kinézet beállításakor is.

3. ábra: A kitöltött adatok oldala

The screenshot shows the 'Egymintás t-próba' dialog box with the following settings:

- Tab: Adatok
- Adatsor ismert:
- Hipotézis típusa: $m \geq m_0$
- Elsőfajú hibavalószínűség (ϵ): 0,05
- m_0 : 25
- Minta jellemzői:
 - elemszám: 12
 - átlag: 24,73
 - szórás: 0,576
- Button: Kilépés

Forrás: a szerző saját szerkesztése.

3.2. Az eredmények megjelenítése

Az adatokat az űrlapon megadva, átlapozhatunk az „Eredmények” lapra. Ekkor az űrlapról az adatok átkerülnek a munkalap celláiba, hogy a munkalapon kiszámítódjon minden eredmény, melyet meg szeretnénk jeleníteni az űrlapon. A kód egy cella kitöltéséhez a következő:

```
Cells(3, 31) = TextBox5.Text
```

Ekkor a munkalap 3. sorának 31. cellájába kerül a TextBox5 beviteli mező adata. Az eredmények űrlapon megjelenítéséhez még annyi szükséges, hogy a munkalapon kialakult eredményeket az űrlapon lévő felirat vezérlőbe helyezzük. Egy ilyen utasítás a következő:

```
Label9.Caption = Cells(12, 17).Text
```

Természetesen ilyen utasításból többre van szükség ahhoz, hogy minden eredmény az egyes cellákból egy-egy felirat vezérlőbe kerüljön.

Az űrlapon nemcsak az eredmények jelennek meg, hanem rendszerezetten a legfontosabb adatokon túl a beállított hipotézis, valamint az egész mondatos válaszok is, ahogy a 4. ábrán látható.

4. ábra: Az eredmények oldala

Egymintás t-próba

Adatok **Eredmények**

Vizsgálendő hipotézis: $m \geq 25$
A próba hipotézisei:
 $H_1: m = 25$
 $H_2: m < 25$

$$t = \frac{\bar{x} - m_0}{s} \cdot \sqrt{n}$$

Adatok	Számítások
$\varepsilon = 0,05$	$t = -1,6238$
$m_0 = 25$	$t(\varepsilon) = -1,7959$
$n = 12$	$p = 0,0664$
átlag = 24,73	
$s = 0,576$	

H_1 elutasítva. ($m < 25$ hamis)
 $\varepsilon = 0,05$ esetén az $m \geq 25$ elfogadható.
A próba szignifikancia szintje: $p = 0,0664$

Kilépés

Forrás: a szerző saját szerkesztése.

Az űrlap bal felső részén először a vizsgálandó hipotézist látjuk, majd a t-próba nullhipotézisét és alternatív hipotézisét. A számítás legfontosabb képletét tájékoztatásként láthatjuk, mint díszítő elemet. A képlet kiszámításához szükséges adatok külön keretben kaptak helyet, mint ahogy a számítások végeredményei is. A statisztikai függvény értéke (t) a munkalapon olyan formulával számított ki, mely hivatkozik az adatokat tartalmazó cellákra:

$$= (AC4 - AC6) / AC5 * AC3^0,5$$

A kritikus tartomány határának munkalap függvénye ($t(\varepsilon)$):

$$= T.INVERZ(AC7; AC3 - 1)$$

A szignifikanciaszint munkalap függvénye (p):

$$= T.ELOSZL(AC8; AC3 - 1; IGAZ)$$

A fenti formulákban a hivatkozott cellák jelentése:

AC3: a minta elemszáma (n).

AC4: a minta átlaga (\bar{x}).

AC5: a minta szórása (s).

AC6: a feltételezett várhatóérték (m_0).

AC7: az elsőfajú hibavalószínűség (ε).

AC8: a statisztikai függvény értéke (t).

Az alsó három sorban egészmondatos válaszokat is kap a felhasználó először az alternatív hipotézis megítéléséről (elutasítva), majd a vizsgálandó ($m \geq 25$) hipotézis elfogadhatóságáról. Végül a szignifikanciaszint (p) fontossága miatt ismét megjelenik.

4. Következtetések

A Microsoft Excel táblázatkezelő program Visual Basic for Applications szolgáltatásával úgy tudtuk kialakítani az egymintás t-próba kiértékelését végző programot, hogy a felhasználó speciális matematikai statisztikai ismeretek nélkül legyen képes használni egyetlen űrlapon, melyen az adatbevitel és az eredmények megjelenítése felhasználóbarát módon történik.

Irodalomjegyzék

- Fabulya Z. (2018a): Access alkalmazás kialakítása ügyfélközpontú szolgáltatások nyilvántartására. *Jelenkori Társadalmi és Gazdasági Folyamatok*, 13 (1-2): 67–76.
- Fabulya Z. (2018b): Access alkalmazás kialakítása dolgozói jelenlét nyilvántartására. *Jelenkori Társadalmi és Gazdasági Folyamatok*, 13 (1-2): 151–160.
- Fabulya Z. (2019): Excel VBA függvények kialakítása háromdimenziós vektorok matematikai alkalmazására. *Jelenkori társadalmi és gazdasági folyamatok*, 14 (1-2): 29–34.
- Fabulya Z. (2020): VBA program fejlesztése feladatsorok dokumentumainak generálására. *Jelenkori társadalmi és gazdasági folyamatok*, 14 (3-4): 139–143.
- Kovalcsik G. (2005): *Az Excel programozása*. Computerbooks, Budapest.
- Matteson B. L. (1995): *Microsoft Excel Visual Basic Programmer's Guide*. MicrosoftPress, Washington.
- Michaletzky Gy., Mogyoródi J. (1995): *Matematikai statisztika*. Nemzeti Tankönyvkiadó, Budapest.
- Móri T. (2011): *Statisztikai hipotézisvizsgálat*. Typotex Kft., Budapest.
- Obádovics J. Gy. (2020): *Valószínűségszámítás és matematikai statisztika*. Scholar Kiadó Kft., Budapest. ISBN: 9789632440675
- Zimmerman M. W. (1996): *Microsoft Office 97 Visual Basic Programmer's Guide*. MicrosoftPress, Washington.