# Csoporttagok:

* Petrik Ádám (GVERV7)
* Balog Barnabás (YRVN14)
* Szilveszter Milán (CZZYLG)
* Gergely Jácint (NPCAPG)

# Regressziós feladat:

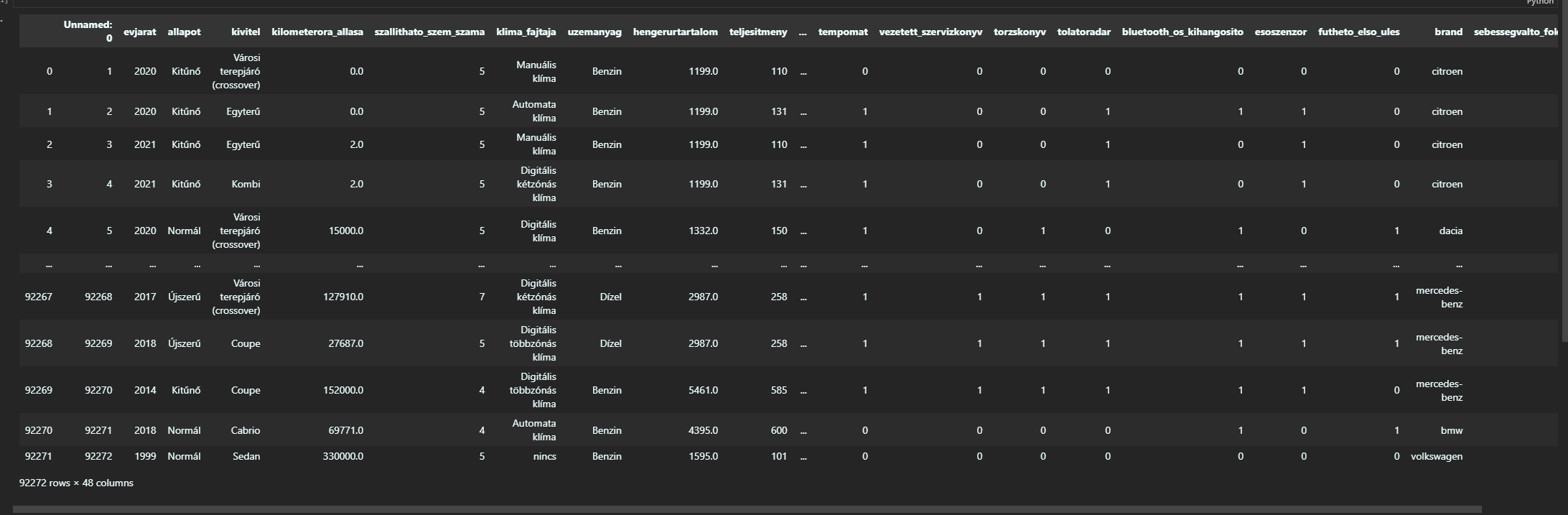
### „Nagy” adathalmaz:

Az adathalmaz a használtautó.hu autókereskedő oldalra feltöltött hirdetések adataiból áll. Az adathalmaz kb. 92 ezer rekordot és 48 attribútumot tartalmaz. A modell célja, hogy az elérhető paraméterekből megbecsüljük az egyes gépjárművek vételárát.

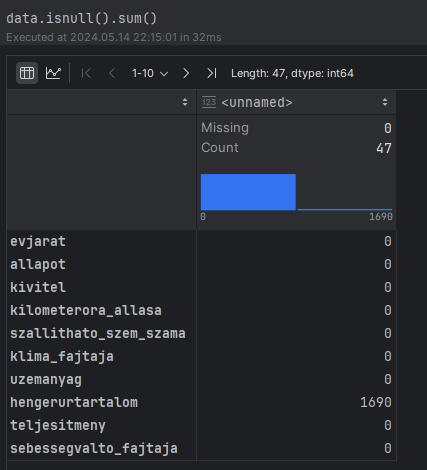
Adattisztítási lehetőségek:

* Karakterkódolási hibák javítása
* Üres értékek helyettesítése
* Szöveges értékek számokká alakítása

**Adatforrás:** <https://www.kaggle.com/datasets/nndorszakmry/hasznltautk-adatai?resource=download>

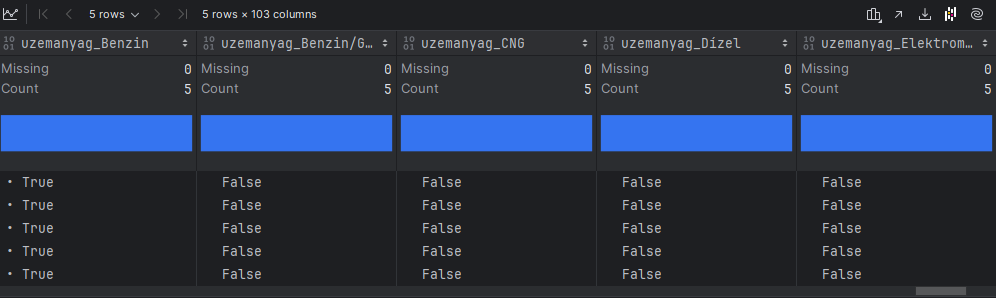
****

Miután beolvastuk az adatokat, ellenőriztük, hogy szükség van-e adattisztításra.

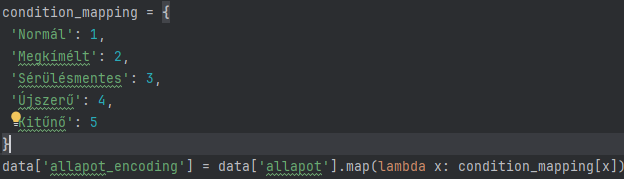


Mivel az elektromos autóknak nincs hengerűrtartalma, így ezeket 0 értékkel töltöttük fel.

A márka oszlopot és az üzemanyag oszlopot felosztottuk külön oszlopokra a **pd.get\_dummies** segítségével. Erre azért volt szükség, mert a regressziós modellek számértékeket várnak input változóként. Ezzel elértük azt, hogy például az üzemanyag típusokra létrejött egy-egy oszlop és amelyik kategóriába tartozik az adott autó ott True az érték. (Ezeket az értékeket már fel tudja ismerni a modell (0/1) és különbséget tud tenni a márkák és üzemanyag típusa között)



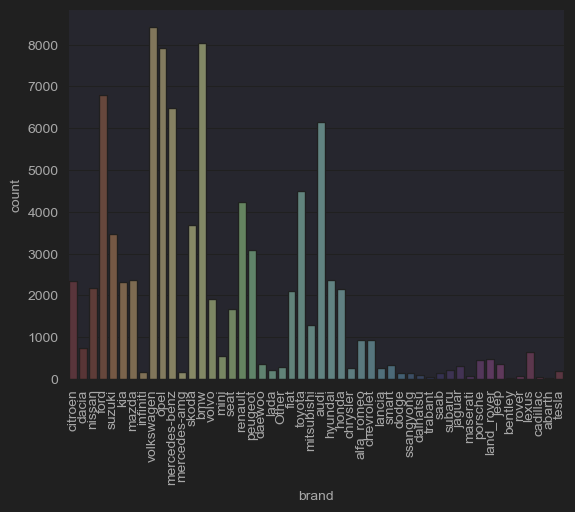
Az autók állapotának is szöveges reprezentációja van az adatok között. Ezeket az értékeket, mivel egy skálát tudunk belőlük alkotni számértékekké konvertáltuk:

****

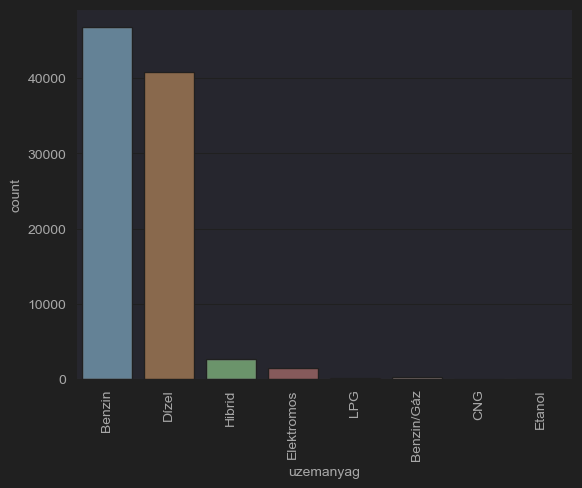
Így olyan számértékeket kaptunk, ahol a szám reprezentálja az állapotot. Minél nagyobb a szám értéke, annál jobb az autó állapota.

**Adatok vizualizációja**

A hirdetett autók darabszáma márkánként



A hirdetett autók darabszáma üzemanyag szerint



**Választott ML algoritmusok:**

* LinearRegression
* Lasso
* Ridge
* DecisionTreeRegressor
* RandomForestRegressor

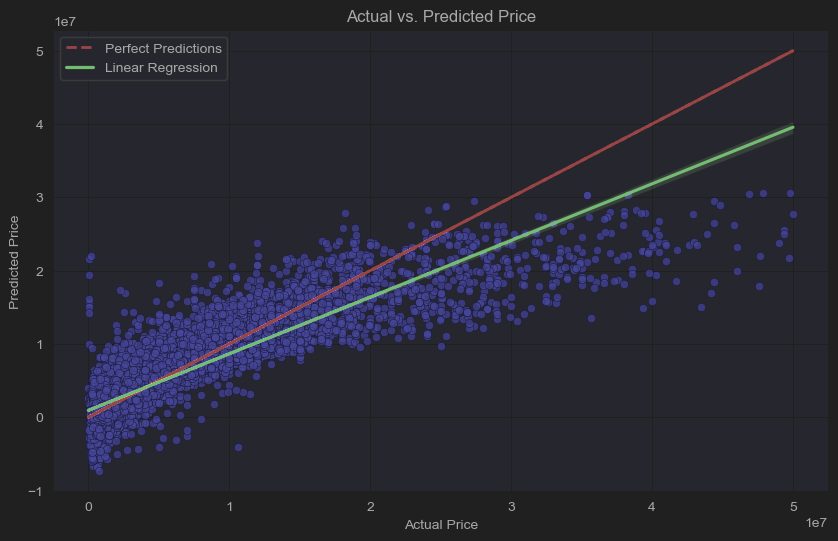
**Tanítás és kiértékelés:**

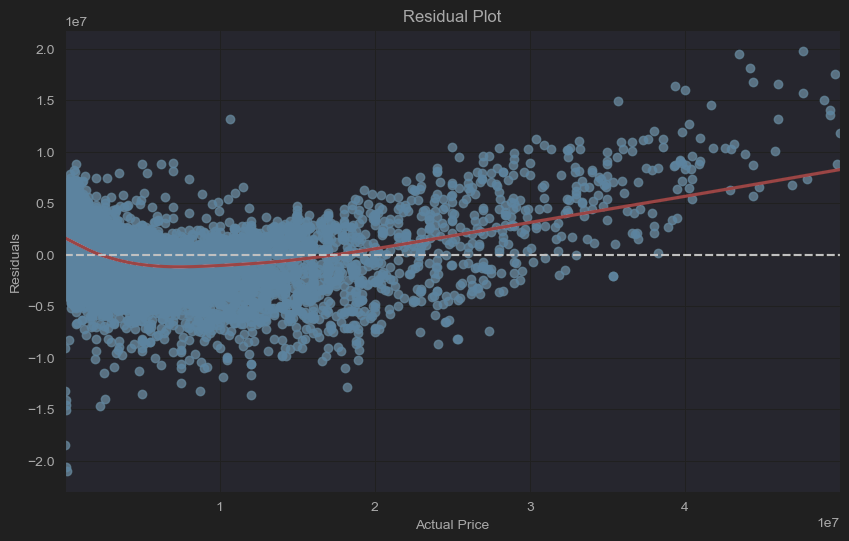
Az adathalmazt felosztottuk tanuló és teszt adatokra az Skicit Learn train\_test\_split() metódusával. Majd a különböző algoritmusokat használva a következő eredményekre jutottunk:

|  |  |
| --- | --- |
| **Eredmények** | |
| **Gépi tanuló algoritmus** | **Pontosság** |
| LinearRegression | 0.7671041674475926 |
| Ridge | 0.7671132515190189 |
| Lasso | 0.7671041606337548 |
| DecisionTreeRegressor | 0.8556583099441392 |
| RandomForestRegressor | 0.8752826786859957 |

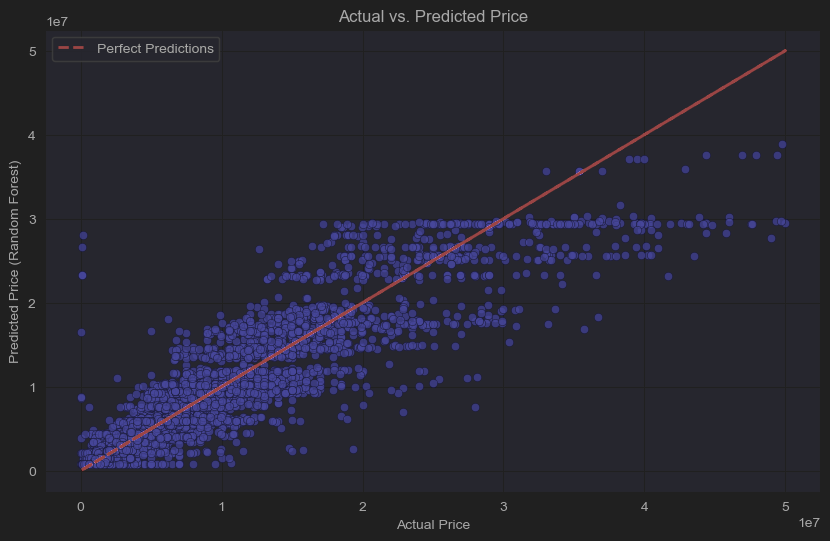
**Eredmények vizualizációja:**

LinearRegression:





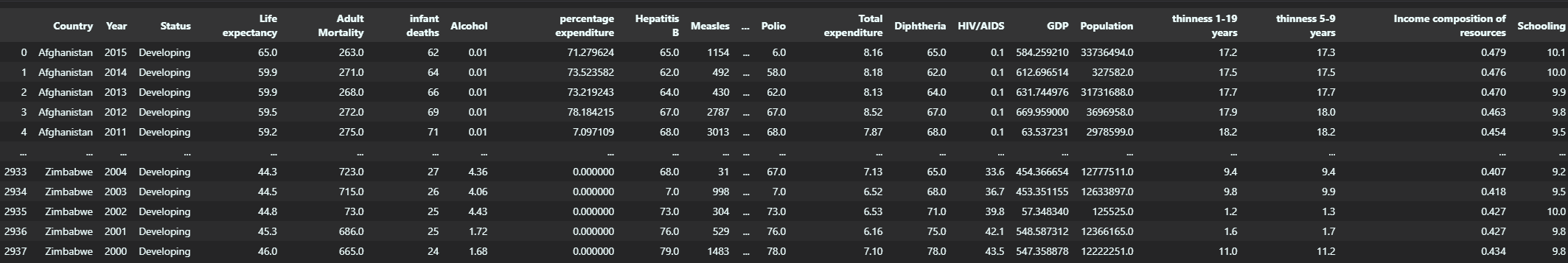
RandomForestRegressor:



### „Kis” adathalmaz:

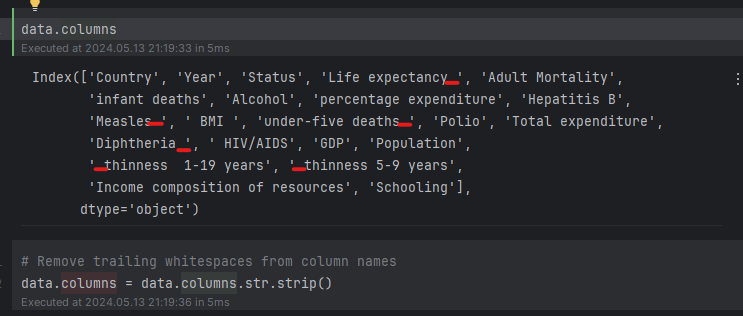
Annak az elemzésére szolgál az adathalmaz, hogy melyek azok a változók, amelyek kulcsfontosságú szerepet játszanak a várható élettartam alakításában. Az adathalmaz közel 3000 rekordot tartalmaz 22 oszloppal.

**Adatforrás**: <https://www.kaggle.com/datasets/kumarajarshi/life-expectancy-who/data>

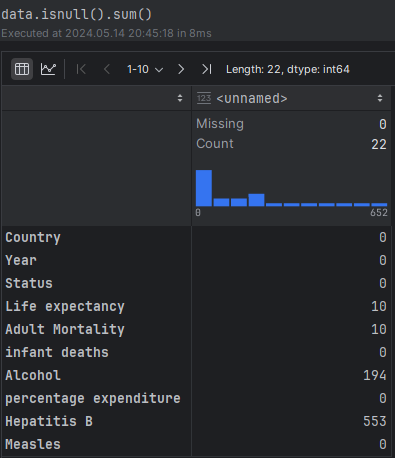


Miután beolvastuk az adatokat, megtisztítottuk őket, hibás adatokat javítottuk.

**Példa:**

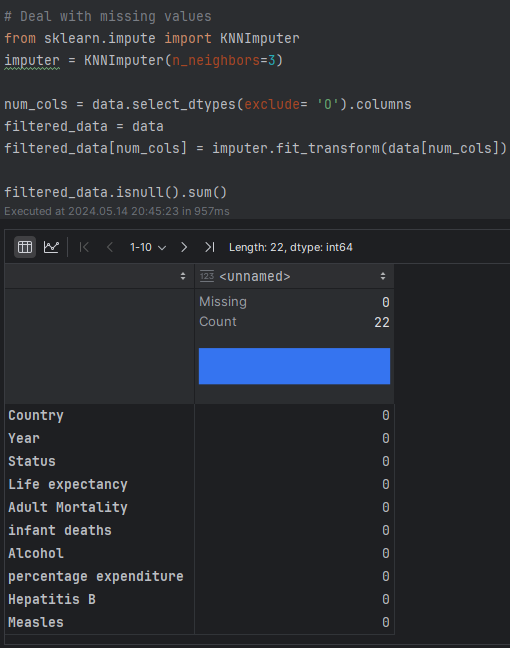


**Hiányzó értékek kezelése:**



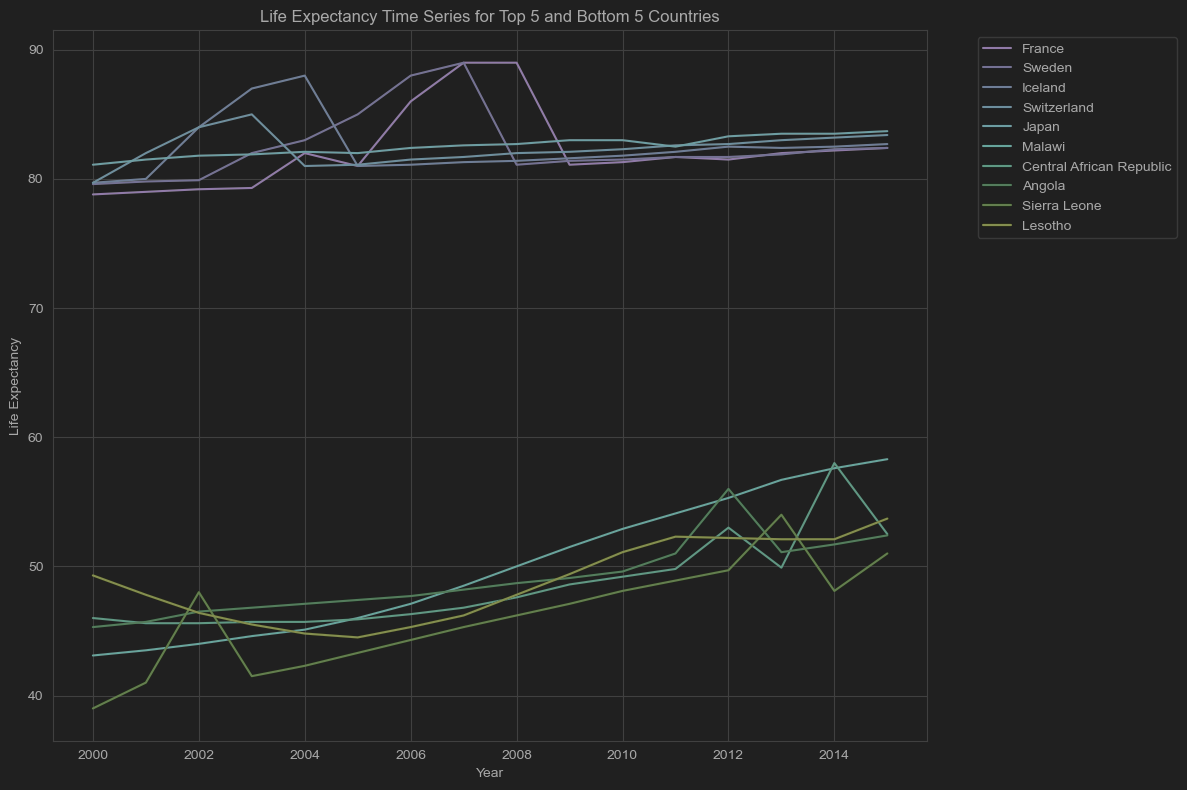
Ha a hiányzó értékek sorait egyszerűen csak törölnénk akkor az adatszettünk a felére csökkenne, így nem lenne reprezentatív az adatunk.

A hiányzó adatok kezelésére az **sklearn.impute** package **KNNImputer** modulját használtuk:

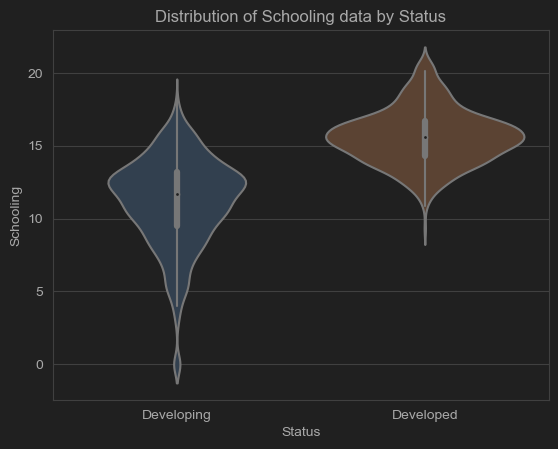


**Adatok vizualizációja:**

Várható élettartam szempontjából az 5 legjobban és legrosszabban teljesítő ország



Iskolázottság eloszlása fejlett és fejlődő országokban:



A diagramok szélessége arra utal, hogy a fejlett országokban jóval magasabb a sűrűsége azoknak, akik több mint 15 év iskolát végeznek, mint a fejlődő országokban.

**Választott ML algoritmusok:**

* LinearRegression
* Lasso
* Ridge
* DecisionTreeRegressor
* RandomForestRegressor

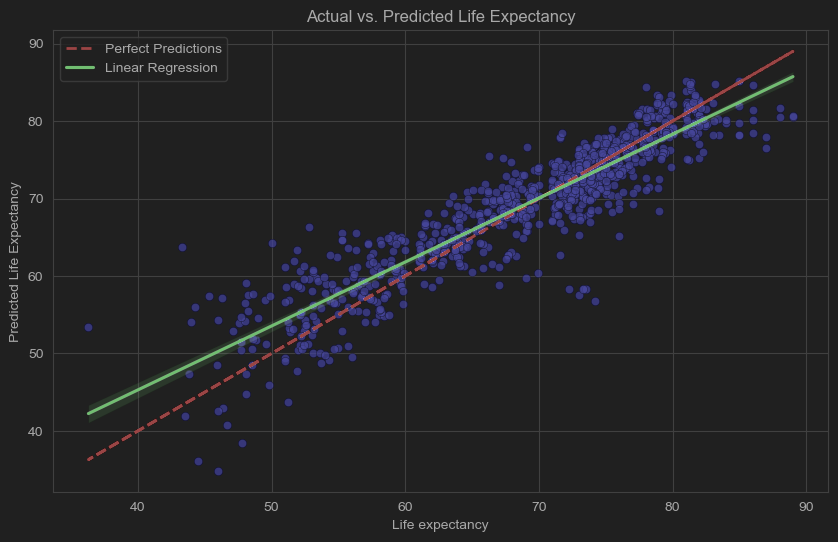
**Tanítás és kiértékelés:**

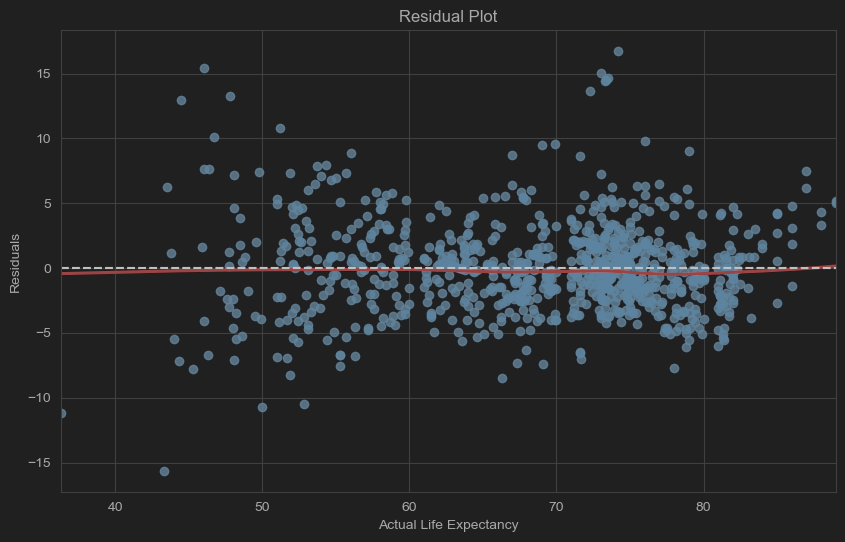
Az adathalmazt felosztottuk tanuló és teszt adatokra az Skicit Learn train\_test\_split() metódusával. Majd a különböző algoritmusokat használva a következő eredményekre jutottunk:

|  |  |
| --- | --- |
| **Eredmények** | |
| **Gépi tanuló algoritmus** | **Pontosság** |
| LinearRegression | 0.8357861997510714 |
| Ridge | 0.8358051000330954 |
| Lasso | 0.8280164129060429 |
| DecisionTreeRegressor | 0.8821965076926579 |
| RandomForestRegressor | 0.9286977571090572 |

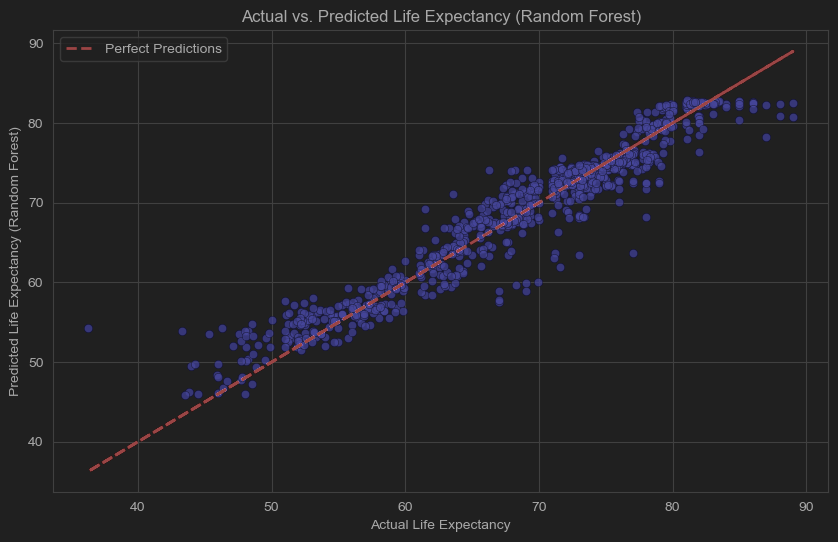
**Eredmények vizualizációja:**

LinearRegression:

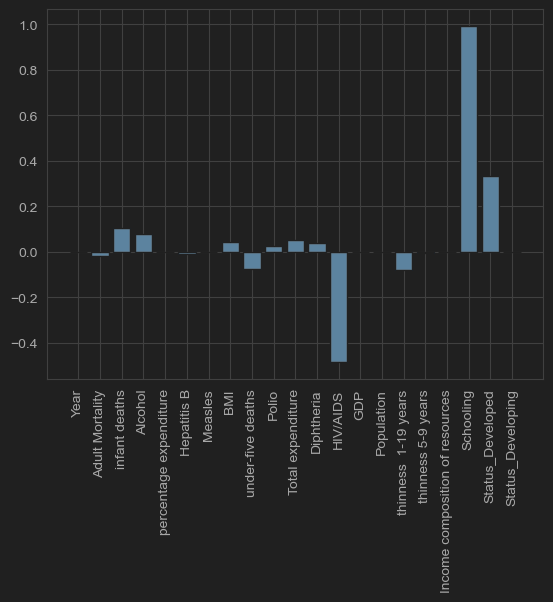




RandomForestRegressor:



Megvizsgáltuk azt is, hogy melyek a főbb befolyásoló feature-ök. az adatszettben. Ezek a tulajdonságok értékei befolyásolják a legjobban a várható élettartamot.

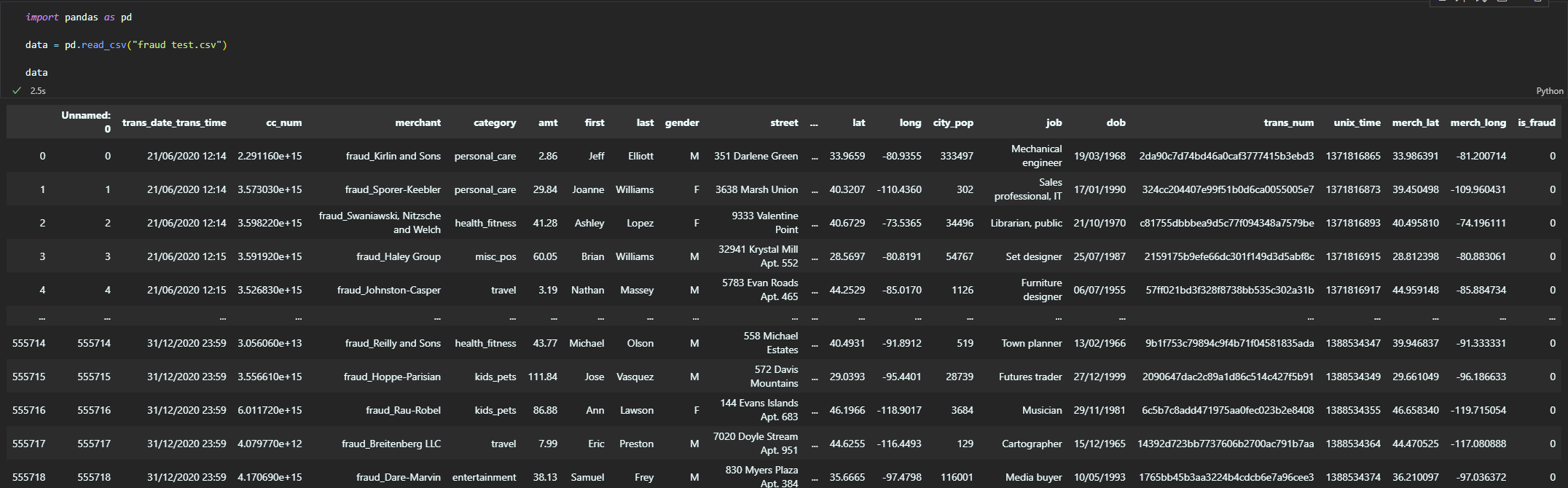


# Osztályozási feladat:

### „Nagy” adathalmaz:

A modell célja a hamis hitelkártya-tranzakciók felderítése, annak érdekében, hogy az ügyfeleket ne terheljék olyan tételekkel, amelyeket nem ők vásároltak. Az adathalmaz több mint 550 ezer rekordot és 23 attribútumot tartalmaz.

**Adathalmaz:** <https://www.kaggle.com/datasets/kelvinkelue/credit-card-fraud-prediction/data>



### „Kis Adathalmaz”:

Ez az adatkészlet az Agaricus és Lepiota családba tartozó 23 kopoltyús gombafajnak megfelelő hipotetikus minták leírását tartalmazza. Minden egyes faj azonosítva van, mint biztosan ehető, biztosan mérgező, vagy ismeretlen ehetőségű és nem ajánlott.

A feltanított modell célja a különböző gombákról eldönteni, hogy mérgezők vagy ehetőek-e.

Az adathalmaz 8123 rekordot és 23 oszlopot tartalmaz.

A képen képernyőkép látható

Automatikusan generált leírás

**Adatforrás**: <https://archive.ics.uci.edu/dataset/73/mushroom>

# Dokumentáció

## Klasszifikáció

### Kis adathalmaz

**Választott ML algoritmusok:**

* Logisztikus Regresszió:
* KNN:
* SVM:
* Random Forest:

**Adatok vizsgálata**

A vizsgálat során kiderült, hogy:

* az adathalmaz oszlopai csak szöveges értékeket tartalmaznak,
* a ’stalk-root’ oszlop 2480 üres értéket tartalmaz,
* az adatok tisztított, összeszedett formában vannak jelen.

Mivel az adathalmazzal különös problémát (sok hiányzó érték, kódolási hiba stb.) nem figyeltem meg, így az adattisztítással nem kell foglalkoznom, hiszen a talált hiányzó értékek nem adatbeviteli hibából adódóan hiányoznak, hanem helyes értékek.

A képen képernyőkép, billentyűzet, számítógép látható

Automatikusan generált leírás

Az adathalmaz első 10 rekordja

**Adatok előkészítése a tanításhoz**

Az adatok előkészítése csak egy lépést foglal magában: az oszlopok értékeit normalizálom úgy, hogy a szöveges értékek helyett szám értékeket rendelek a *Scikit Learn* könyvtár *LabelEncoder* osztályának használatával.

A normalizálás után az adatok az alábbi módon néznek ki:

A képen képernyőkép, billentyűzet, számítógép látható

Automatikusan generált leírás

Adathalmaz a Label Encoding után

*Feature Scaling*-et nem szükséges alkalmazni, mert az oszlopok így is eléggé kis varianciával rendelkeznek.

**Tanítás és kiértékelés**

Az adathalmazt tanítás előtt szétbontottuk tanító és tesztelő adathalmazokra a *Scikit Learn train\_test\_split()* metódusának segítségével, majd először minden algoritmust az alapvető beállításaival „tanítottam” fel.

|  |  |
| --- | --- |
| **Eredmények** | |
| **Gépi tanuló algoritmus** | **Pontosság** |
| Logistic Regression | 95,57% |
| KNN (k=1) | 99,95% |
| SVM | 100% |
| Naív Bayes | 93,11% |
| Decision Tree | 100% |
| Random Forest | 100% |

### Nagy adathalmaz

**Adatok vizsgálata**

asd

**Adatok előkészítése a tanításhoz**

asd

**Tanítás és kiértékelés**