1. **Adatfelderítés**: Töltsétek le a Titanik adatbázist (<https://www.kaggle.com/c/titanic/data>) elég lesz csak a train adatbázis. Ellenőrizzétek a hiányzó értékeket és kezeljétek a problémát. Értsétek meg, hogy mik a változók, csináljatok pár plot (legalább 2-t), ami segíti a megértést (pl ki vesz 1st class jegyet?).
2. **LOGREG**: Használjatok egy logisztikus regressziót a túlélők meghatározására. Használjatok 5 fold CV-t (shuffle-t is). Fontos hogy 5 fold CV-t használjunk vagy elég lenne a sima train-test split?
3. **Döntési fa**
   1. Elsőnek minden feature nélkül prediktáljátok azt, hogy ki éli túl.
   2. Osszátok szét az adatot random módon 70-15-15 train-val-test set-re. Vizsgáljátok meg a döntési fa viselkedését, úgy hogy legalább 3 paramétert változtattok (class\_weightet itt még ne használjátok). Készítsetek plotokat, amik megmutatják a train,val,test accuracy-t a különböző paraméter specifikációk kapcsán. Jó ötlet a validációs adatra támaszkodni a hyper paraméterek hangolása kapcsán? Vűálasszátok ki a legjobban teljesítő specifikációt.
   3. Végül pruningoljátok az eredeti tree-t cost complexity pruninggal
   4. Hasonlítsátok össze a 3 fát. Értelmezzétek az eredményket (overfittingre térjetek ki), válasszátok ki a legjobban teljesítő modellt. Rajzoljátok fel a fát az sklearn-ös és a dtreeviz-es packegek segítségévelés értelmezzétek pár modatban.
4. **Modell összehasonlítás**
   1. Hasonlítsátok össze a logisztikus regresszió és a legjobb döntési fa eredményét. Confusion mátrixok, roc görbe, PR görbe (görbéket egy ábrán ábrázoljátok), nézzéteki meg a különböző metrikákat: AUC, ACC és még két másikat legalább. Melyik modellt választanátok?
   2. Most változtassátok meg a döntési fa class\_weight paraméterét, egyszer úgy, hogy a túlélők predikciója, egyszer úgy hogy a halottak predikciója a hangsúlyosabb.
   3. Hasonlítsátok össze és értelmezzétek a különbségeket az előbbi 3 fa kapcsán (Confusion matrix, ROC, PR, acc, auc score stb.).
5. **Szorgalmi** (sör+unikum kombóért cserébe). Implementáljátok a DecTreeClassifiert csak numpyban. Hasonlítsátok össze az eredményeket az sklearn packeggel. Nyílván fenn van az interneten :), szóval itt a kód bármely részletébe belekérdezhetek és el kell magyarázni, hogy az mit csinál pontosan.