Tema Învățare Automată

Partea 1

Dată publicare: 27/10/2025	Dată limită rezolvare: 21/11/2025

1. Descriere generală

În practica de zi cu zi a unui inginer sau cercetător în domeniul învățării automate intră frecvent următoarele trei aspecte:

- Vizualizarea și “explorarea” datelor unei probleme (Exploratory Data Analysis)
- Încercarea de a extrage atribute ale datelor problemei pentru a fi utilizate în obiectivul de analiză ales (e.g. clasificare, regresie, detectie de anomalii)
- Evaluarea mai multor modele pentru găsirea soluției celei mai bune pentru problema dată

Sarcinile voastre de lucru vor solicita utilizarea de biblioteci de **vizualizare a datelor (crearea de diagrame)**, **extragerea de atribute (feature extraction)** pentru folosirea algoritmilor de clasificare discutați la curs, precum și **utilizarea unor modele** de machine learning.

2. Descrierea Seturilor de Date

Aveți la dispoziție două seturi de date:

- **Închirierea bicicletelor** - un set de date ce conține informații despre închirierea de biciclete pe o perioadă de doi ani, cu date orare. Scopul este de a prezice numărul total de biciclete închiriate în fiecare oră din setul de testare.
- **Autovit** - un set de date extras de pe site-ul autovit.ro privind caracteristicile autoturismelor listate spre vânzare pe site. Scopul este de a prezice prețul de vânzare al autoturismului. Setul de date este mai provocator în prelucrare, având multiple coloane cu date lipsă, coloane cu valori numerice, ordinale sau categorice.

Seturile de date sunt în format CSV și au fost deja împărțite în date de antrenare (train) și de testare (test).
Seturile de date se descarcă de pe platforma Moodle, fiind atașate enunțului temei.

3. Regresie cu cuantile

Regresia cu cuantile (Quantile Regression) este o extensie a regresiei liniare clasice care nu mai încearcă să prezică valoarea medie a unei variabile ţintă, ci anumite *cuantile* (percentile) ale distribuției acesteia. De exemplu, în loc să estimeze doar „valoarea aşteptată" a vânzărilor viitoare, un model de regresie cu cuantile poate prezice valoarea sub care se vor afla 10% din cazuri (cuantila 0.1), valoarea mediană (0.5) sau cea sub care se află 90% dintre observaţii (0.9). Această abordare este utilă mai ales în prognoze cu incertitudine sau în situaţii în care erorile nu sunt distribuite simetric, cum ar fi în date economice, meteorologice sau în cazul setului de date Autovit (de tip retail).

În cadrul scikit-learn, regresia cu cuantile poate fi implementată cu clasa **QuantileRegressor** (sau, în cazul de regresie de tip ensemble, folosind **GradientBoostingRegressor** cu **loss="quantile"**).
Un exemplu detaliat de folosire a regresorului GradientBoosting cu un loss de tip quantile este prezentat în acest tutorial (Prediction Intervals for Gradient Boosting Regression).
În exemplu se antrenează modele cu **alpha=0.05**, **0.5** şi **0.95** pentru loss="quantile". Aceste trei modele permit estimarea unei **benzi de predicţie** (de exemplu între cuantila 0.05 şi 0.95 → o bandă de ~90% predicţie).

În cod:
gbr_low = GradientBoostingRegressor(loss="quantile", alpha=0.05, …).fit(X_train, y_train)
gbr_med = GradientBoostingRegressor(loss="quantile", alpha=0.50, …).fit(X_train, y_train)
gbr_high = GradientBoostingRegressor(loss="quantile", alpha=0.95, …).fit(X_train, y_train)

Predicţiile gbr_low.predict(X_test) şi gbr_high.predict(X_test) delimitează intervalul de predicţie.

În utilizarea regresiei cu cuantile există câteva **atenţionări practice**. Regresorul cu cuantile este mai lent decât o regresie liniară obișnuită, deoarece optimizează o funcţie de pierdere neliniară ("pinball loss") şi soluţia nu are formă analitică simplă.
În cazul modelului **GradientBoosting** pentru cuantile, trebuie acordată atenţie calibrării intervalului de predicţie: exemplul din scikit-learn arată că pe date de test acoperirea observată a intervalului 90% (între cele două cuantile 0.05 şi 0.95) a fost de ~86,8% în loc de 90%. Trebuie ținut cont că hiperparametrii optimi pentru cuantila de jos (alpha=0.05) sau de sus (alpha=0.95) pot fi diferiţi de cei folosiţi pentru mediana (alpha=0.5), deci tuning-ul separat pentru fiecare este indicat (a se vedea exemplul din tutorial de aplicare a unei proceduri de Cross-Validation pentru cautarea hiperparametrilor).

Pentru vizualiazrea rezultatelor unei predictii pe cuantile, se pot raporta tabele cu predicția la nivelul alpha_low, alpha_med, alpha_high și MSE (mean squared error) sau se pot face grafice de vizualizare a predicției. A se vedea secțiunea de analiză a erorilor din tutorial.

4. Cerințe

4.1. Explorarea și Vizualizarea datelor [2p]

Efectuați o analiză a exploratorie a datelor (a se vedea primul laborator) care să vă aducă informații asupra complexității problemei.
Surprindeți prin **tabele** și **grafice (diagrame)** aspecte ce țin de:

- Vizualizare ca serie de timp a datelor din închirierea bicicletelor - existență trend-uri sau ciclicitate
- Corelații cu target-ul, corelații între atribute
- Existența datelor lipsă

Atenție! Alegeți tipul de explorare / vizualizare cu folos. Punctele pe cerință se obțin în urma unei justificări clare în textul raportului a motivului pentru care ați ales acel tip de analiză.
Sunt cerute cel puțin 4 tipuri de analiză per dataset pentru punctaj complet.

4.2. Extragerea, standardizarea, selecția de atribute și suplimentarea valorilor lipsă [3p]

Seturile de date au o mixtură de tipuri de atribute (categorice, numerice), cu valori numerice de ordine diferite de mărime sau valori lipsă.
Urmăriți documentația scikit-learn referitoare la următoarele:

- Standardizarea atributelor
- Encodarea atributelor categorice sau ordinale
- Discretizarea atributelor numerice
- Imputarea valorilor lipsă
- Selecția atributelor relevante

Analizați opțiunile **necesare** și **favorabile** seturilor voastre de date.
Atenție! Documentați în raportul temei pașii pe care îi efectuați, împreună cu o justificare pentru metodele de preprocesare alese. Justificarea voastră trebuie să facă referire și la rezultatele obținute.

4.3. Utilizarea algoritmilor de Învățare Automată [5p]

Pentru efectuarea taskului de clasificare peste fiecare set de date veți folosi următorii algoritmi:

- LogisticRegression - folosiți implementarea din scikit-learn
- SVR - folosiți implementarea din scikit-learn
- RandomForest Regressor - folosiți implementarea din scikit-learn
- GradientBoosted Regressor - folosiți implementare din scikit-learn
    - Antrenare cu loss = “squared error”
    - Antrenare cu loss = “quantile” (a se vedea indicațiile din Secțiunea 3)
- Quantile Regressor - folosiți implementarea din scikit-learn

Fiecare algoritm din cei propuși are o serie de **hiper-parametri** care influențează funcționarea acestuia. Pentru a găsi valorile potrivite pentru aceștia veți folosi o procedură de **căutare a hiper-parametrilor.**
Setul minim de hiper-parametri de căutat este:

- LogisticRegression - parametru *C* de regularizare, metodologia de clasificare multinomială (parametrul multi_class ales între “ovr” sau “multinomial”)
- SVR: tipul de kernel, parametru *C* de regularizare
- RandomForest: numărul de arbori, adâncimea maximă a unui arbore, procentul din input folosit la antrenarea fiecărui arbore
- GradientBoostedRegressor: numărul de arbori, adâncimea maximă a unui arbore, learning rate, hiperparametrii specifici pentru quantile regression.
- Quantile Regressor: parametrul *alpha* de regularizare L1

Căutarea hiper-parametrilor se poate face în prin două metode:

- Folosind un **set de validare**
- Folosind procedura de **Randomized Search with Cross-Validation**

Atenție! Procedura de căutare prin Cross Validation **poate dura foarte mult**, dacă setul de date este mare, alegeți multe fold-uri și căutați după mulți parametri. Analizați dacă puteți restructura seturile voastre de date, astfel încât să obțineți un **set de validare** pe care să-l folosiți pentru procedura de potrivire a hiper-parametrilor, renunțând astfel la Cross-Validation.

Evaluarea algoritmilor
În raportul vostru trebuie să prezentați următoarele:

- Rezultatul procedurii de preprocesare, etapele efectuate și justificarea acestora.
- Pentru fiecare algoritm, realizați un tabel în care să prezentați **media și varianța** pentru **MSE, MAE, R^2 score** sau **quantile predictions** (pentru regresorii de cuantila) la nivelul fiecărui set de date
    - Pe linii va fi indexată configurația de hiper-parametri rezultată din procedura de căutare
    - Pe coloane vor fi prezentate metricile cerute
    - **Relevați prin bolduire** valorile maxime pentru fiecare metrică

5. Predarea temei

Tema va fi încărcată pe Moodle însoțită de un raport sub formă de fișier PDF, care include:

- **Cerința 4.1** - cuprinde toate vizualizările și statisticile cerute. **Este obligatorie** prezența în text a **unei interpretări / analize** a diagramelor rezultate.
- **Cerințele 4.2-4.4** - include raportarea preprocesării de atribute și a evaluării algoritmilor de clasificare pentru cele două seturi de date propuse. **Este obligatorie** prezența în text a **unei interpretări / analize** a rezultatelor obținute (e.g. care atribute sunt cele mai predictive, cât de puternic este impactul hiper-parametrilor asupra performanței fiecărui algoritm considerat, o analiză a erorilor de predicție observate).

Rezultatele temei vor fi prezentate în cadrul laboratoarelor de Învățare Automată, **exclusiv pe baza rapoartelor încărcate.**

6. Anexa

A. Setul de date “Inchiriere de biciclete”:

1) **Structura datelor:**
   Set de antrenament (train.csv): Primele 19 zile din fiecare lună
   Set de testare (test.csv): Zilele 20 până la sfârșitul fiecărei luni

   Trebuie să preziceți numărul total de biciclete închiriate în fiecare oră acoperită de setul de testare, folosind doar informații disponibile înainte de perioada de închiriere.

2) **Coloanele Setului de Date:**

Coloana	Tip	Descriere
data_ora	DateTime	Data și ora (format: YYYY-MM-DD HH:MM:SS)
sezon	Integer	Sezonul anului: • 1 = primăvară • 2 = vară • 3 = toamnă • 4 = iarnă
sarbatoare	Binary	Indică dacă ziua este considerată sărbătoare legală: • 0 = nu • 1 = da
zi_lucratoare	Binary	Indică dacă ziua nu este nici weekend, nici sărbătoare: • 0 = nu (weekend sau sărbătoare) • 1 = da (zi lucrătoare)
vreme	Integer	Condițiile meteorologice: • 1 = Senin, Puțini nori, Parțial înnorat • 2 = Ceață + Înnorat, Ceață + Nori sparți, Ceață + Puțini nori • 3 = Zăpadă ușoară, Ploaie ușoară +

		Furtună + Nori sparți • 4 = Ploaie torențială + Grindină + Furtună + Ceață, Zăpadă + Ceață
temperatura	Float	Temperatura în grade Celsius
temperatura_resimtita	Float	Temperatura "resimțită" (feels like) în grade Celsius
umiditate	Integer	Umiditatea relativă (%)
viteza_vant	Float	Viteza vântului
ocazionali	Integer	Numărul de închirieri inițiate de utilizatori neînregistrați (doar în train.csv)
inregistrati	Integer	Numărul de închirieri inițiate de utilizatori înregistrați (doar în train.csv)
total	Integer	**ȚINTA** - Numărul total de închirieri de biciclete

3) Întrebări frecvente:

Î: Care este diferența dintre *ocazionali* și *inregistrati*?
R: *ocazionali* reprezintă utilizatorii fără cont (walk-in customers), iar *inregistrati* sunt utilizatorii cu abonament sau cont. Suma lor dă *total*.

Î: De ce coloanele *ocazionali* și *inregistrati* nu există în test.csv?
R: Acestea sunt parte din informația viitoare pe care trebuie să o preziceți. În scenarii reale, nu cunoașteți dinainte câți utilizatori vor închiria biciclete.

B. Setul de date “Autovit”:

1) **Structure datelor**:
   - `train_cars_listings.csv` - setul de antrenament, conținând aproximativ **19.000** de anunțuri.
   - `val_cars_listings.csv` - setul de validare/testare, conținând aproximativ **4.700** de anunțuri.

   Obiectivul este reprezentat de estimarea prețului de vânzare al unui autoturism pe baza unor caracteristici relevante.

2) **Coloanele setului de date**:

Coloana	Tip	Descriere
nume	String	Titlul anunțului publicat de vânzător (ex: "BMW X5 3.0d M-Package 2018"). Coloană de tip metadată.
pret	Float	Prețul de listare al mașinii (în euro); **variabila țintă** (care trebuie prezisă).
Oferit de	String	Tipul vânzătorului: dealer auto sau persoană fizică.
Are VIN (Serie sasiu)	Binary	Indică dacă anunțul include seria de șasiu (VIN).
Marca	String	Marca producătorului (ex: "Volkswagen", "BMW", "Dacia")
Model	String	Modelul mașinii (ex: "Golf", "X5", "Logan")
Versiune	String	Versiunea sau echiparea specifică a modelului.
Anul fabricației	Integer	Anul în care a fost fabricată mașina
Km	Float	Kilometrajul total parcurs (în kilometri)
Combustibil	String	Tipul de combustibil: benzină, motorină, hibrid, electric etc.
Putere	Float	Puterea motorului exprimată în cai putere (CP).
Capacitate cilindrica	Float	Capacitatea motorului (în cm^3)
Transmisie	String	Tipul de tracțiune: față, spate, integrală (4x4)

		etc.
Consum Extraurban	Float	Consum mediu declarat în afara localității (l/100 km).
Cutie de viteze	String	Tipul cutiei de viteze (manuală, automată etc.)
Consum Urban	Float	Consum mediu declarat în oraș (l/100 km).
Tip Caroserie	String	Tipul caroseriei (Sedan, SUV, Coupe etc.)
Emisii CO2	Float	Nivelul emisiilor de dioxid de carbon (g/km)
Numar de portiere	Integer	Numărul de uși ale vehiculului.
Culoare	String	Culoarea caroseriei.
Numar locuri	Integer	Numărul total de locuri disponibile.
Garantie dealer (inclusa in pret)	Integer	Indică dacă mașina este vândută cu garanție inclusă oferită de dealer (luni).
Primul proprietar (de nou)	Binary	Specifică dacă vânzătorul actual este primul proprietar al mașinii.
Fara accident in istoric	Binary	Indică dacă mașina nu a fost implicată în accidente.
Stare	Binary	Starea generală a vehiculului (nou, second-hand)
Audio si tehnologie	List[String]	Dotări audio și tehnologice (ex: sistem navigație, Bluetooth).
Confort si echipamente optionale	List[String]	Dotări de confort (ex: climă automată, scaune încălzite, piele).
Electronice si sisteme de asistenta	List[String]	Sisteme electronice și de asistență (ex: pilot automat, sistem start/stop).
Siguranta	List[String]	Dotări de siguranță (ex: airbag-uri, ABS, ESP).
Norma de poluare	String	Norma de emisii a motorului (ex: Euro 5, Euro 6).
Tara de origine	String	Țara de proveniență a vehiculului.
Data primei inmatriculari	Date (String)	Data primei înmatriculări a mașinii.
Garantie de la producator pana la	Date (String)	Data până la care garanția de producător este valabilă.
Vehicule electrice	List[String]	Dotări specifice vehiculelor electrice/hybride (ex: sistem recuperare energie, funcție încărcare rapidă, cablu de încărcare).

Tuning	Binary	Indică dacă mașina are modificări estetice sau tehnice neoriginale.
Autonomie	Integer	Autonomia declarată (în kilometri) pentru vehiculele electrice/hibride.
Capacitate baterie	Integer	Capacitatea bateriei (în kWh) pentru vehiculele electrice.