

Proiect

Identificarea sistemelor 2022-2023

Organizare

Acest proiect este parte obligatorie a cursului de Identificarea Sistemelor, seria B, linia de licență Ingineria Sistemelor, Universitatea Tehnică din Cluj-Napoca. Nota la proiect are o pondere de 30% în nota finală a disciplinei (15% partea 1, și 15% partea 2). Proiectul va fi efectuat în grupuri de câte **trei** studenți, și rezolvarea sa necesită aproximativ 20 ore / student, depinzând și de experiența fiecăruia în MATLAB. Fiecare grup va primi seturile sale de date separate. Pentru a le primi, formați cât mai repede grupuri și trimiteți un email îndrumătorului de proiect Zoltán Nagy, la adresa `zoltan.nagy@aut.utcluj.ro`. Menționați numele și adresele de email ale tuturor membrilor grupului.

Proiectul constă din două probleme. Evaluarea se face diferit pentru cele două părți, vezi mai jos fiecare parte pentru detalii. **Regulă esențială:** Este strict interzisă copierea de cod, text, sau rezultate de la colegi sau din alte surse cum ar fi website-uri etc. Verificările de plagiarism se fac cu metode automate, și nu se va tolera nici un fel de material copiat: încălcarea acestei reguli duce imediat la pierderea dreptului de participare la examen. Fiți așadar extrem de atenți!

Partea 1. Modelarea seriilor de timp cu funcții de bază Fourier

Vom considera o serie de timp conținând cantitățile lunare vândute dintr-un anumit produs de către un magazin. Magazinul vinde mai ales componente de instalații, cum ar fi țevi, conectori, boilere, etc. Există așadar produse care se vând mai ales toamna, cum ar fi boilerele, fiindcă clienții se pregătesc pentru sezonul rece, și alte produse care se vând mai ales primăvara, cum ar fi sistemele de irigație. În plus, există produse care se vând din ce în ce mai mult de-a lungul timpului, cum ar fi materialele de izolare termică. Datele înregistrează timpul (în luni) și numărul de unități vândute din produsul relevant, de-a lungul unei durate de timp de mai mulți ani.

Setul de date este furnizat sub forma unui fișier de date MATLAB, conținând doi vectori de dimensiuni identice: `time`, unde fiecare element conține indexul k al lunii, și `y`, care conține cantitatea de produs $y(k)$ corespunzătoare fiecărei luni.

Înainte de modelare, setul de date trebuie împărțit în două – o parte de identificare și o a doua de validare. Primele 80% dintre eșantioane se vor folosi pentru identificare, și ultimele 20% pentru validare.

Vom crea modele ale seriei de timp de forma:

$$\hat{y}(k) = t_0 + t_1 k + \sum_{i=1}^m \left[a_i \cos\left(\frac{2\pi i k}{P}\right) + b_i \sin\left(\frac{2\pi i k}{P}\right) \right]$$

Formula include atât o componentă de *tendință liniară*, de ordinul întâi $t_0 + t_1 k$, cât și o *bază Fourier* cu un număr configurabil de termeni m . Folosim o bază Fourier pentru că ne așteptăm ca datele să manifeste periodicitate (anuală, trimestrială etc.) În particular, avem date lunare și vom presupune o periodicitate cel mult anuală, așadar perioada implicită va fi $P = 12$. Fiecare termen Fourier generează două funcții de bază, \cos și \sin . Regresorii acestui model conțin componentele ce reprezintă tendința liniară (1 și k) precum și funcțiile de bază Fourier (\cos și \sin de diverse frecvențe). Vectorul de parametri este $\theta = [t_0, t_1, a_1, b_1, \dots, a_m, b_m]^T$, și conține $2 + 2m$ valori în total.

De exemplu, pentru $m = 1$ și perioada aleasă $P = 12$, aproximatorul are forma:

$$\hat{y}(k) = t_0 + t_1 k + a_1 \cos\left(\frac{2\pi k}{12}\right) + b_1 \sin\left(\frac{2\pi k}{12}\right)$$

$$= \varphi^\top(k)\theta = \left[1, k, \cos\left(\frac{2\pi k}{12}\right), \sin\left(\frac{2\pi k}{12}\right)\right] \cdot \begin{bmatrix} t_0 \\ t_1 \\ a_1 \\ b_1 \end{bmatrix}$$

iar pentru $m = 2$:

$$\hat{y}(k) = t_0 + t_1 k + a_1 \cos\left(\frac{2\pi k}{12}\right) + b_1 \sin\left(\frac{2\pi k}{12}\right) + a_2 \cos\left(\frac{4\pi k}{12}\right) + b_2 \sin\left(\frac{4\pi k}{12}\right)$$

$$= \varphi^\top(k)\theta = \left[1, k, \cos\left(\frac{2\pi k}{12}\right), \sin\left(\frac{2\pi k}{12}\right), \cos\left(\frac{4\pi k}{12}\right), \sin\left(\frac{4\pi k}{12}\right)\right] \cdot \begin{bmatrix} t_0 \\ t_1 \\ a_1 \\ b_1 \\ a_2 \\ b_2 \end{bmatrix}$$

Pentru o valoare m dată, găsirea modelului constă din aflarea vectorului optim de parametri θ^* pentru care \hat{y} se apropie cel mai mult de y pe setul de date de identificare, în sensul celor mai mici pătrate. Parametrii optimi se pot afla folosind regresia liniară. Detaliile se găsesc în curs, Partea 2: *Baze matematice*, în secțiunea de regresie liniară.

Raportul trebuie scris coerent, concis și autosuficient. Ar trebui să includă cel puțin următoarele elemente:

- O parte introductivă, care include o descriere a problemei.
- O scurtă descriere a structurii aproximatorului și a procedurii de găsire a parametrilor.
- Orice caracteristici esențiale ale soluției dvs. individuale (nu includeți detalii triviale de implementare).
- Rezultate de acordare (cel puțin MSE în funcție de m , sub formă de grafic sau de tabel).
- Graficele reprezentative menționate mai sus, pentru valoarea optimă a lui m .
- Discuția menționată mai sus și o concluzie generală.

Un ghid mai detaliat privind redactarea și stilul raportului în general este disponibil pe site.

Evaluare partea 1

Soluția proiectului constă dintr-un scurt raport, scris în engleză sau în română, și din codul asociat. Un grup produce un singur raport. Termenul limită pentru raport și cod este **13 noiembrie 2022, cel târziu la 24:00**. În caz că un grup întârzie cu trimiterea soluției, fiecare nouă zi de întârziere începută duce la pierderea a două puncte din nota maximă (de exemplu, dacă soluția este livrată în 15 noiembrie 2019 la ora 00:10AM, nota maximă devine 6 fiindcă a început deja a doua zi de întârziere).

Vă rugăm să **fiți atenți la, și să urmăriți strict**, următoarele instrucțiuni pentru trimiterea soluției. Un proces uniform și semi-automat pentru procesarea soluțiilor este esențial pentru eficiența în notare. Orice deviație de la reguli înseamnă că soluția va necesita timp adițional pentru procesare manuală, și cum acest timp în general nu este disponibil, riscați să nu vi se poată nota soluția.

- Soluția va consta din exact două fișiere, denumite exact: Nume1Nume2Nume3.pdf și Nume1Nume2Nume3.zip, unde Numei este numele de familie al studentului “i” din grupul de proiect, fără diacritice. De exemplu: IonescuFarkasBonta.pdf și IonescuFarkasBonta.zip.
- Primul fișier este raportul, care trebuie livrat **în format PDF** (nu DOCX sau orice alt format sursă; doar PDF).
- Pentru identificare, raportul trebuie să conțină pe prima pagină numele studenților din grup și indicele setului de date pentru grup.
- Raportul trebuie să includă listing-uri complete ale codului MATLAB dezvoltat (funcții și scripturi).
- Al doilea fișier conține codul în sine, trimis separat într-o **arhivă ZIP** (nu RAR, nu 7Zip sau alte formate; doar ZIP clasic). Important: Arhiva nu are voie să conțină subdirectoare, toate fișierele .m trebuie să fie în rădăcina arhivei.
- Aceste fișiere se vor trimite via un DropBox file request, al cărui link va fi trimis înainte de deadline. Nu trimiteți versiuni multiple, acestea nu vor fi luate în considerare. Doar prima variantă va fi luată în considerare, asigurați-vă așadar că este completă și corectă.
- Data de creare a fișierului în sistemul DropBox este folosită pentru calculul penalităților de întârziere conform regulilor de mai sus.

Partea 2. ARX neliniar

Pentru a efectua această parte din proiect, trebuie să aveți bazele despre modele ARX *liniare* din materialul de curs, partea *Metode ARX*. Metoda ARX neliniară este explicată și ea într-o anexă a aceleiași părți.

Se dă un set de date măsurat de la un **sistem dinamic** cu o intrare și o ieșire. Ordinul sistemului nu este mai mare de 3, și dinamica poate fi neliniară, în timp ce ieșirea poate fi afectată de zgomot. Vom dezvolta un model de tip cutie neagră pentru acest sistem, folosind o structură ARX neliniară de tip polinom de semnalele de intrare și ieșire anterioare. Un al doilea set de date, măsurat de la același sistem, este furnizat pentru validarea modelului dezvoltat. Cele două seturi de date sunt furnizate într-un fișier MATLAB, respectiv în variabilele `id` și `val`, ambele obiecte de tip `iddata` din toolbox-ul de identificarea sistemelor. Reamintim că intrarea, ieșirea, și perioada de eșantionare sunt disponibile în câmpurile `u`, `y`, `Ts` ale acestor obiecte. Doar în caz că toolbox-ul nu este instalat, aceleși seturi de date sunt furnizate și în format vectorial, `id_array` și `val_array`, fiecare dintre ele o matrice cu structura: valorile de timp pe prima coloană, intrarea pe a doua, și ieșirea pe ultima coloană.

Un model ARX neliniar cu ordinele na , nb , și întârzierea nk , folosind aceeași convenție ca și funcția MATLAB `arx`, are structura:

$$\begin{aligned} \hat{y}(k) &= p(y(k-1), \dots, y(k-na), u(k-nk), u(k-nk-1), \dots, u(k-nk-nb+1)) \\ &= p(d(k)) \end{aligned} \quad (1)$$

unde vectorul de ieșiri și intrări întârziate este notat $d(k) = [y(k-1), \dots, y(k-na), u(k-nk), u(k-nk-1), \dots, u(k-nk-nb+1)]^T$, și p este un polinom de gradul m de aceste variabile.

De exemplu, dacă $na = nb = nk = 1$, atunci $d = [y(k-1), u(k-1)]^T$, și dacă luăm gradul $m = 2$, putem scrie explicit polinomul:

$$y(k) = ay(k-1) + bu(k-1) + cy(k-1)^2 + vu(k-1)^2 + wu(k-1)y(k-1) + z \quad (2)$$

unde a, b, c, v, w, z sunt coeficienți reali și parametrii modelului. De notat că modelul este neliniar, conține pătrate și produse între variabilele întârziate (spre deosebire de ARX standard, care ar conține doar termeni liniari în $y(k-1)$ and $u(k-1)$). Modelul (1) are o proprietate esențială: este liniar în parametri, ceea ce înseamnă că parametrii pot fi găsiți folosind metoda regresiei liniare.

De notat că forma liniară ARX este un caz special al formei generale (1), obținut prin alegerea gradului $m = 1$, ceea ce duce la:

$$\hat{y}(k) = ay(k-1) + bu(k-1) + c$$

și în plus impunând ca termenul liber să fie $c = 0$ (fără această condiție, modelul se numește afin).

Cerințele sunt descrise în cele ce urmează. Programați o funcție care generează un model ARX neliniar de tip polinomial, pentru ordine na, nb și gradul m configurabile; nk poate fi lăsat egal cu 1. Programați de asemenea procedura de regresie pentru identificarea parametrilor, și utilizarea modelului cu intrări noi. De notat că această utilizare trebuie să se poată face în două moduri:

- Predicție (cu un pas înainte), care folosește valorile reale ale ieșirilor întârziate ale sistemului; în exemplul (2), la pasul k s-ar aplica ecuația (2) folosind variabilele $y(k-1)$ și $u(k-1)$ în partea dreaptă a egalității.
- Simulare, în care ieșirile precedente ale sistemului nu sunt disponibile, și ca atare nu se pot folosi decât ieșirile anterioare ale modelului însuși; în exemplu, $y(k-1)$ ar fi înlocuit cu valoarea simulată precedent $\hat{y}(k-1)$ în partea dreaptă a ecuației (2)

Identificați un astfel de model ARX neliniar folosind setul de date de identificare, și validați-l pe setul de validare. Acordați atenție ordinea modelului, precum și gradul polinomului, pentru a obține o performanță cât mai bună pe datele de identificare. Pentru a simplifica procedura de căutare, puteți lua $na = nb$. Prezentarea dvs. ar trebui să includă cel puțin următoarele elemente:

- O parte introductivă, care include o descriere a problemei.
- O scurtă descriere a structurii aproximativului și a procedurii de găsire a parametrilor.
- Orice caracteristici esențiale ale soluției dvs. individuale (nu includeți detalii triviale de implementare).
- Rezultate de reglare (cel puțin MSE în funcție de $na = nb$ și de m , fie ca tabel, fie ca grafic).
- Pentru cel mai bun model obținut mai sus, grafice reprezentative cu ieșirea aproximată a modelului comparată cu ieșirea reală, atât pentru simulare, cât și pentru predicție, separat pentru seturile de date de antrenare și de validare.
- Corespunzător acestor grafice, raportați eroarea de predicție și eroarea de simulare atât pentru setul de identificare, cât și pentru cel de validare (folosiți eroarea medie pătratică).
- O discuție a rezultatelor, inclusiv calitatea modelului pe cele două seturi de date; și o concluzie generală.

Prezentarea de mai sus este suficientă pentru implementarea metodei, dar pentru detalii tehnice puteți studia de ex. următoarele articole:

1. H. Peng et al., *RBF-ARX model-based nonlinear system modeling and predictive control with application to a NOx decomposition process*, Control Engineering Practice 12, paginile 191–203, 2007. Modelul este explicat în secțiunile 2.1-2.2, dar folosește în loc de polioname funcții de bază radiale reglabile în poziție și lățime.
2. L. Ljung, *System Identification*, Wiley Encyclopedia of Electrical and Electronics Engineering, 2007. Disponibil sub formă de raport tehnic cu numărul LiTH-ISY-R-2809. Vezi secțiunea 4 pentru modele neliniare, care din nou folosesc funcții de bază.

Evaluare partea 2

Partea 2 va fi prezentată oral, și codul trebuie de asemenea trimis. Prezentările vor avea loc la finalul semestrului, vezi programul de pe site. Programul exact (intervalul orar în care prezintă fiecare grup) va fi comunicat cu un timp suficient în avans. Fiecare grup are 15 de minute, din care 8 sau 9 sunt pentru prezentarea în sine, și minim 6 pentru întrebări. Fiți prezenți cu 15 minute în avans. Toți membrii grupului trebuie să participe și să răspundă la întrebări, și cu excepția situațiilor de forță majoră, orice student absent nu va avea proiectul validat, și ca atare nu va fi eligibil pentru examen. Nota va lua în considerare trei aspecte: soluția în sine (cod, rezultate); prezentarea; și răspunsurile la întrebări. Scopul principal al întrebărilor va fi diferențierea contribuției fiecărui student din grup, deci în general fiecare student va primi altă notă.

Termenul limită pentru prezentare și pentru cod este **23 decembrie 2022, cel târziu la 24:00**. În caz că un grup întârzie cu trimiterea soluției, fiecare nouă zi de întârziere începută duce la pierderea a două puncte din nota maximă, ca și pentru partea 1.

Vă rugăm să urmăriți, la fel de strict ca și pentru partea 1, următoarele instrucțiuni pentru trimiterea soluției.

- Soluția va consta din cel puțin două fișiere, denumite exact: Nume1Nume2Nume3.pdf; și Nume1Nume2Nume3.zip, unde Numei este numele de familie al studentului “i” din grupul de proiect. De exemplu: IonescuFarkasBonta.pdf și IonescuFarkasBonta.zip.
- Primul fișier este prezentarea, care trebuie livrată **în format PDF**. Opțional, se poate trimite și o variantă PPT sau PPTX, care va fi folosită în timpul sesiunii dacă este posibil. Indiferent de format, este nevoie de o prezentare (slide-uri) – nu trimiteți raport în locul prezentării!
- Pentru identificare, primul slide al prezentării trebuie să conțină proeminent numele studenților din grup și indicii grupului, sub forma N/M unde N este indexul fișierului de date pentru partea 1, și M pentru partea 2.
- Al doilea fișier conține codul în sine, trimis separat într-o **arhivă ZIP** (nu se acceptă alte formate). Important: Arhiva nu are voie să conțină subdirectoare, toate fișierele .m trebuie să fie în rădăcina arhivei.
- Aceste fișiere se vor trimite via un DropBox file request, al cărui link va fi trimis înainte de deadline. Nu trimiteți versiuni multiple, acestea nu vor fi luate în considerare. Doar prima variantă va fi luată în considerare, asigurați-vă așadar că este completă și corectă.
- Data de creare a fișierului în sistemul DropBox este folosită pentru calculul penalităților de întârziere conform regulilor de mai sus.

Programare în MATLAB și remarci adiționale

Dacă sunteți mai puțin familiari cu programarea în MATLAB, următoarele idei vă pot fi de folos. Scrieți `doc` în linia de comandă pentru accesarea documentației. Un nod util la început este *Getting Started with MATLAB*. Vor fi de folos de asemenea și *Matrices and Arrays*, *Programming Basics*, și *Plotting Basics*.

Faceți un efort pentru a produce un cod MATLAB compact și elegant, evitând folosirea buclelor `for`, `while` și a instrucțiunilor `if-then-else` pentru operații care pot fi efectuate mai simplu folosind vectori sau matrici. Indicii în această direcție găsiți căutând conceptul de “vectorizare” (en. *vectorization*) în documentația MATLAB. Nu exagerați însă cu vectorizarea: dacă programul este mai clar sau ușor de înțeles cu bucle sau instrucțiuni `if`, folosiți-le.