



M Ű E G Y E T E M 1 7 8 2

**Budapesti Műszaki és Gazdaságtudományi Egyetem**  
Villamosmérnöki és Informatikai Kar  
Távközlési és Médiainformatika Tanszék

Borkó Károly Gusztáv

**VILLAMOSENERGIA-  
FOGYASZTÁS ELŐREJELZÉSE  
GÉPI TANULÁSI  
MÓDSZEREKKEL**

KONZULENS

**Gáspár Csaba**

BUDAPEST, 2022

# Tartalomjegyzék

<b>Összefoglaló .....</b>	<b>5</b>
<b>Abstract.....</b>	<b>6</b>
<b>1 A villamosenergia rendszer működése.....</b>	<b>7</b>
1.1 A villamosenergia rendszer szereplői .....	8
1.1.1 Termelők.....	8
1.1.2 Szállítók .....	9
1.1.3 Kereskedők .....	10
1.1.4 Felhasználók .....	10
1.1.5 Szabályozó hatáság.....	11
<b>2 Magyarország áramfogyasztási adatainak elemzése .....</b>	<b>12</b>
2.1 Éves átlagos fogyasztás elemzése .....	12
2.2 Havi átlagos fogyasztás elemzése .....	13
2.3 Heti átlagos fogyasztás elemzése.....	15
2.4 Napi átlagos fogyasztás elemzése .....	16
<b>3 Idősoros adatok feldolgozásának módszerei .....</b>	<b>17</b>
3.1 Gépi tanulási módszerek problémái.....	18
<b>4 Gépi tanulási módszerek .....</b>	<b>19</b>
4.1 Gépi tanulási modellek hatékonyságának mérése .....	19
4.2 Döntési fák.....	20
4.3 Ensemble módszerek .....	21
<b>5 Az implementáció során felhasznált technológiák.....</b>	<b>23</b>
5.1 A Python nyelv .....	23
5.2 A Numpy könyvtár .....	24
5.3 A Pandas könyvtár .....	24
5.4 A Scikit-Learn könyvtár .....	25
5.5 A Dateutil könyvtár .....	26
5.6 A Matplotlib könyvtár .....	26
<b>6 Áramfogyasztás előre jelző modellek bemutatása .....</b>	<b>27</b>
6.1 Mavir Zrt. adathalmazának előfeldolgozása .....	28
6.2 Bemeneti változók fontosságának vizsgálata .....	29
6.3 Az alapmodell bemutatása .....	30

6.4 Hőmérsékletet alkalmazó modell.....	32
6.5 Munkarendet alkalmazó modell.....	34
6.6 Késleltetett fogyasztást alkalmazó modell.....	36
6.7 Modellek összegzése.....	37
<b>7 A modell interpretációja .....</b>	<b>39</b>
7.1 A PDP grafikonok használata modellek interpretációjához .....	40
7.2 Áramfogyasztási modellek értelmezése.....	40
<b>8 Összefoglalás és továbbfejlesztési lehetőségek .....</b>	<b>45</b>
<b>Irodalomjegyzék.....</b>	<b>46</b>

# HALLGATÓI NYILATKOZAT

Alulírott Borkó Károly Gusztáv, hallgató kijelentem, hogy ezt a TDK dolgozatot meg nem engedett segítség nélkül, saját magam készítettem, csak a megadott forrásokat (szakirodalom, eszközök stb.) használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Hozzájárulok, hogy a jelen munkám alapadatait (szerző(k), cím, angol és magyar nyelvű tartalmi kivonat, készítés éve, konzulens(ek) neve) a BME VIK nyilvánosan hozzáférhető elektronikus formában, a munka teljes szövegét pedig az egyetem belső hálózatán keresztül (vagy hitelesített felhasználók számára) közzétegye.

Kelt: Budapest, 2022. 10. 31.

.....  
Borkó Károly Gusztáv

# Összefoglaló

A villamos-energetikai rendszerek üzemeltetőinek egyik legfontosabb feladata, hogy biztosítsák a hálózat zavartalan működését. Ehhez az szükséges, hogy fenntartsák a termelés és a fogyasztás egyensúlyát. A rendszerszabályozás hatékony végrehajtásához az üzemeltetőknek jó hatásfokkal előre kell tudniuk jelezni mind a termelés, mind a fogyasztás várható értékét a következő időszakra. A fogyasztás előrejelzése az energia-kereskedők számára is fontos, mivel a várható fogyasztási adatok birtokában kedvezőbb kereskedői szerződéseket tudnak kötni a termelőkkel és a szállítókkal. Magyarországon az átviteli hálózat üzemeltetését és a rendszerszabályozást a MAVIR Zrt. végzi. Ez a szervezet nyilvánosan publikálja az országos fogyasztási adatokat.

Jelen TDK dolgozatomban a villamosenergia-fogyasztás előrejelzésével foglalkozom. Azt a kérdést kívánom megválaszolni, milyen pontossággal lehet előre jelezni a fogyasztási adatokat. Munkám során a MAVIR Zrt. nyilvános hálózati rendszerterhelési adatait használtam fel. Ez az adathalmaz 15 percenként összegezve az elmúlt 10 év országos áramfogyasztási adatait tartalmazza. Dolgozatomban először röviden áttekintem az elektromos áramhálózat működését, majd a Magyarországi villamosenergiafelhasználás sajátosságait. Ismertetem az idősoros adatok feldolgozásának főbb módszereit, kiemelten az általam is használt gépi tanulási módszereket. Bemutatom a gépi tanulási algoritmusok ensemble családját, különösen az általam használt RandomForest algoritmus működését. Definiálom a megoldott konkrét problémát és kitérek az általam készített algoritmusok hatékonyságának értékelési módszereire is.

Ezt követően iteratívan mutatom be az általam megalkotott gépi tanulási modelleket és algoritmusokat. A legegyszerűbb és legrosszabb hatékonyságú modellektől haladok a bonyolultabb és hatékonyabb modellek felé. Minden bemutatott modell az azt megelőző gépi tanulási megoldás egy továbbfejlesztett optimalizált változata. Minden modell hatékonyságát több kiértékelési metrika segítségével jellemzek. Legvégül pedig értékelem az elért eredményeket és jövőbeli továbbfejlesztési lehetőségeket fogalmazok meg.

# Abstract

One of the most important tasks of the operators of electricity systems is to ensure the smooth operation of the network. For this, it is necessary to maintain a balance between production and consumption. In order to implement system regulation effectively, operators must be able to predict the expected value of both production and consumption for the next period with a good degree of efficiency. Forecasting consumption is also important for energy traders, as they can enter into more favorable trading contracts with producers and suppliers when they have the expected consumption data. In Hungary, MAVIR Zrt. operates the transmission network and regulates the system. This organization publicly publishes national consumption data.

In my present TDK thesis, I deal with the forecasting of electricity consumption. During my work, I used the public network system load data of MAVIR Zrt. This data set contains the national electricity consumption data of the last 10 years summarized every 15 minutes. In my thesis, I will first briefly review the operation of the electricity network, and then the specifics of electricity use in Hungary. I will describe the main methods of processing time series data, especially the machine learning methods I also use. I will present the ensemble family of machine learning algorithms, especially the operation of the RandomForest algorithm that I used. I define the specific problem and I also discuss the methods of evaluating the effectiveness of the algorithms I made.

After that, I will iteratively present the machine learning models and algorithms I created. I progress from the simplest and least efficient models to more complex and more efficient models. Each model presented is an improved and optimized version of the previous machine learning solution. I characterize the effectiveness of each model using several evaluation metrics. Finally, I evaluate the results achieved and formulate future development opportunities.

# 1 A villamosenergia rendszer működése

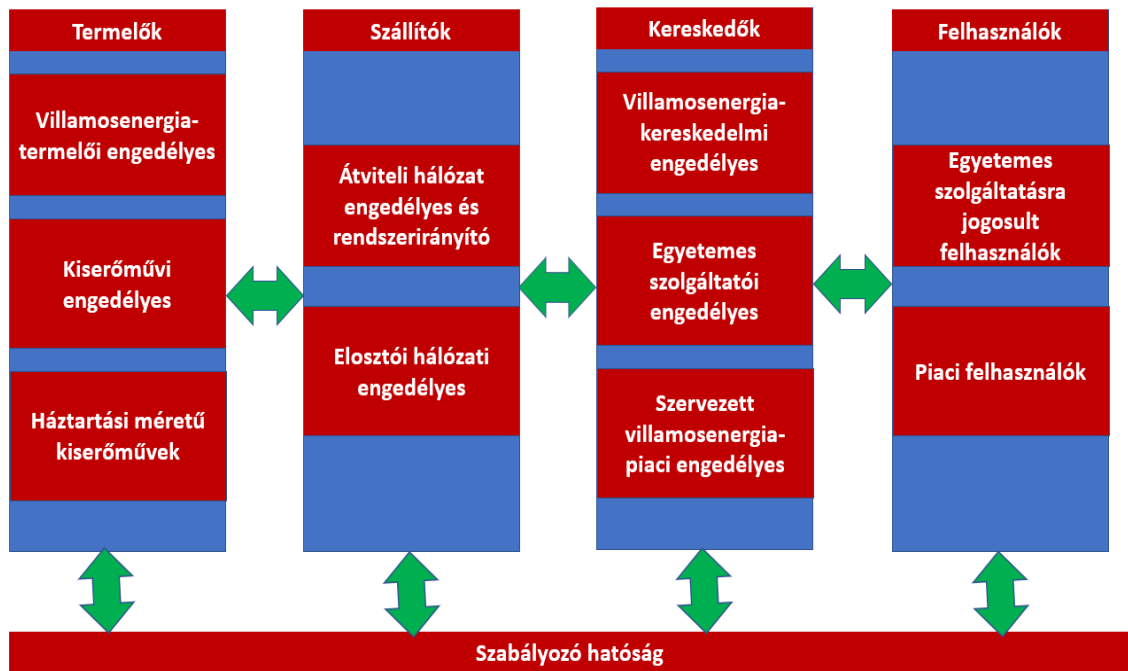
Minden ország számára az egyik legfontosabb nemzetstratégiai kérdés a biztonságos áramellátás. Egyik ország sem engedheti meg magának, hogy a villamosenergia rendszerében tartós zavar lépjen fel, amelynek következtében a rendszer leáll és hosszú ideig nem áll rendelkezésre elektromos áram. Az elektromos hálózat stabil és megbízható működtetése nem egyszerű feladat, a fenntartás előrelátó szakszerű tervezést és szabályozott piaci környezetet igényel. A legtöbb ország nem képes önerőből fedezni az energiaigényeket, ilyenkor más országoktól vásárol villamos áramot, ezért fontos, hogy a villamosenergiarendszer jól együtt tudjon működni más országok rendszereivel és a nemzetközi áramkereskedelemmel fedezni tudják az átmeneti energiahiányt vagy értékesíteni tudják az energiátöbbletüket. [1]

A villamosenergia hálózat stabil és megbízható működésének kulcsa a hálózat villamos teljesítményének az egyensúlya. A rendszer működése során minden időpillanatban szükséges, hogy a betáplálások és a fogyasztások egyensúlyban legyenek. Ez biztosítja, hogy az elektromos áram teljesítménye, feszültsége és frekvenciája viszonylag állandó legyen, ne mutasson nagy ingadozásokat. A hálózat működtetését még az a tényező is nehezíti, hogy jelen pillanatban a villamos energia tárolása nem megoldott. Mivel korlátozott hatékonyságú szivattyús tárolók állnak csak rendelkezésre, az átmeneti fogyasztói ingadozások miatt fellépő eltérő energiaigényeket nem tudják tartalékokkal fedezni, azt csupán erőművi termelés által biztosítható a fogyasztók számára. [1]

A villamosenergia rendszer stabilitásának biztosítása az ellátó hálózat üzemeltetőjének a feladata. Az elektromos áram szállítása a termelőktől a fogyasztókig egyetlen hálózati infrastruktúrán keresztül történik. Az átviteli hálózat stabilitását rendszerszabályozással biztosítják. A rendszer terhelése, vagyis a villamosenergia felhasználás az év különböző időszakában nem állandó. Ezáltal a rendszerszabályozást végzők az igényekhez igazítják a betáplálásokat, vagyis a termeléseket. Ezen kívül mindig fenntartanak biztonsági termelési tartalékokat, amely segítségével az átmeneti stabilitási problémákat orvosolják. Ezt úgy oldják meg, hogy üzemzavar esetén ezt a biztonsági tartalékot a rendszerbe pumpálva stabilizálják a hálózatot. [2]

## 1.1 A villamosenergia rendszer szereplői

A villamosenergia rendszereknek 5 fő szereplője van. Ezek a szereplők a termelők, szállítók, kereskedők, felhasználók és az ezeket összekapcsoló szabályozó hatóság. A következőkben bemutatom az egyes érintetteket és ezen csoportok főbb jellemzőit.



1.1. ábra: Villamosenergia rendszer érintettjei és kapcsolódási pontjaik [1]

### 1.1.1 Termelők

Az érintettek első nagy csoportját a termelők alkotják. Ezen csoport magában foglalja a villamosenergiát előállító entitásokat. Ide tartoznak a villamosenergia-termelői engedéllyel rendelkező 50MW feletti elektromos teljesítményű nagyerőművek, a kiserőművi engedéllyel rendelkező 50kW és 50MW közötti teljesítményű kiserőművek, valamint az 50kW teljesítmény alatti háztartási méretű, általában a lakossági szereplők által működtetett kiserőművek.

Ezen entitások feladata a villamosenergia előállítása. A szereplők, mint eladók jelennek meg az elektromos áram kereskedelmében, akik a megtermelt áramot a felhasználóknak értékesítik a szállítók és a kereskedők közreműködésével. A megtermelt elektromos áramot a szállítók segítségével juttatják el a felhasználókig. [2]



## 1.1.2 Szállítók

A másik nagy csoport a szállítók csoportja. A szállítókat hálózati engedélyeseknek is nevezik. A szállítók felelnek azért, hogy a megtermelt elektromos áram az átviteli és az elosztói hálózaton keresztül eljusson a fogyasztókig. Az átviteli hálózat felelős az erőműveknél megtermelt villamosenergia szállításáért. Az átviteli hálózatra csatlakoznak az elosztói hálózatok, amik az átviteli hálózatot a fogyasztók csatlakozási pontjaival kötik össze. A szállítói engedélyesek működési költségeit rendszerhasználati-díj néven fizetik meg a felhasználók. [2]

Az átviteli hálózatot az átviteli rendszerirányítók, vagyis a Transmission system operator-ok, röviden a TSO-k üzemeltetik. Magyarországon az átviteli hálózat üzemeltetője és egyben tulajdonosa a Magyar Villamosenergia-ipari Átviteli Zrt., röviden MAVIR Zrt. Az átviteli rendszerirányítók feladata az átviteli hálózat üzemeltetése, karbantartása és fejlesztése. Az ő felelőségük, hogy a hálózatukon továbbításra kerülő elektromos áram megfelelő minőségben áramoljon keresztül az általuk üzemeltetett hálózaton. A feladatkörükbe tartozik még az átviteli és elosztói hálózatok karbantartásának és fejlesztésének a menedzsmentje.

A TSO-k felelnek továbbá a villamosenergia rendszer biztonságos működtetéséért. Ez megáiban foglalja a rendszerszabályozási feladatokat, az üzembiztonsági szolgáltatásokat, az üzemelőképzítéseket és tervezéseket, valamint a nemzetközi összeköttetések kialakítását a belföldi hálózattal. [1]

A fent említett műszaki feladatok mellett az átviteli rendszerirányítók ezen műveletek adminisztrációs és pénzügyi és jogi kérdéseivel is foglalkoznak. Az ő felelőségük a termelők és a kereskedők közötti szerződések megkötése, a hálózat kapacitásainak aukciójának lebonyolítása, elszámolási rendszer működtetése, a kötelező adatszolgáltatások menedzsmentje, valamint a nemzetközi energiapiaci integráció feladatainak ellátása. [3]

Az elosztói hálózatokat az elosztó hálózati engedélyesek, a Distribution Network Operator-ok röviden DSO-k üzemeltetik. A DSO-k fő feladata, hogy a villamos energiát eljuttassák a fogyasztói csatlakozási pontokra. Feladataik közé tartozik még a fogyasztói mérőberendezések működtetése, valamint a fogyasztási adatokkal kapcsolatos adatszolgáltatások teljesítése a TSO-k felé. Magyarországon hat hálózati elosztó üzemel.

### **1.1.3 Kereskedők**

A villamosenergiapiac szereplőinek harmadik csoportja a kereskedők csoportja. A végfelhasználókat a kereskedők szolgálják ki. Ők kötik össze a végfogyasztókat a szállítókkal. A kereskedőket 3 fő csoportba sorolhatjuk.

Az első csoportot a szabad villamosenergia-piac alkotja, melyben a kereskedők és a fogyasztók piaci áron egymással megállapodva tudnak az elektromos árammal kereskedni. Ebben az esetben a kereskedők birtokolják a felhasználó mérőberendezései feletti kontrollt. Ezen kereskedelmi piacon lehetőség nyílik a megvásárolt elektromos áram tovább értékesítésére is. [1]

A második csoportot az egyetemes szolgáltatók alkotják. Az egyetemes szolgáltatás a szabad-piac mellett működik. Az egyetemes szolgáltatók az erre jogosult végfelhasználóknak egy kedvezőbb hatósági áron értékesítik az elektromos áramot. Ezen szegmens résztvevői nem kereskedhetnek a villamosenergiával, azt csak, mint végfogyasztók használhatják fel. Magyarországon egyetlen kereskedő lát el egyetemes szolgáltatást, ez a Magyar Villamos Művek Next Energiakereskedelmi Zrt. [3]

A harmadik kereskedelmi csoport az úgynevezett szervezett villamosenergia piac, vagyis az áramtőzsde. A tőzsde tagjai szabályozott vételi és eladási megbízásokat adhatnak meg. A kereskedés anonim módon történik, az árat és a mennyiséget a kereslet és a kínálat határozza meg. Magyarországon egyetlen szabályozott áramtőzsde létezik, ez a HUPX. [2]

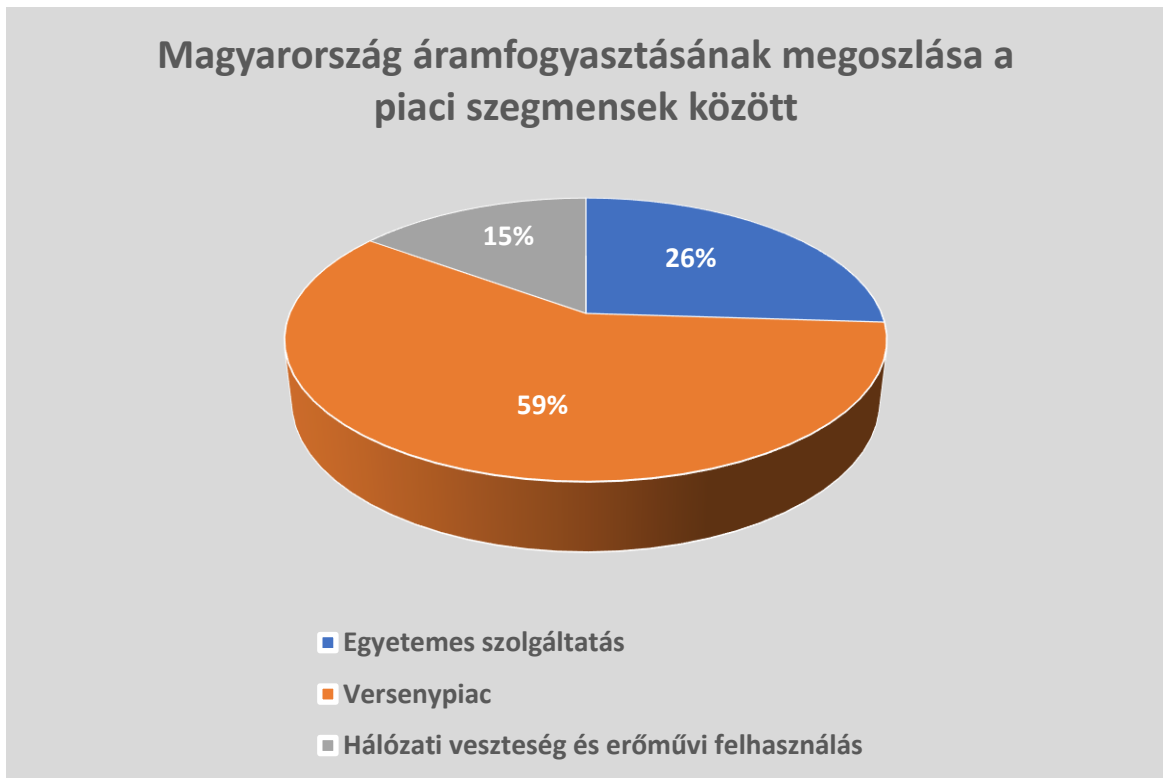
### **1.1.4 Felhasználók**

A villamosenergiapiac szereplőinek negyedik csoportját a felhasználók, vagyis a fogyasztók alkotják. A felhasználók használják fel az erőművek által megtermelt villamos energiát. A fogyasztókat három fő csoportba sorolhatjuk. [4]

1. Egyetemes szolgáltatásra jogosultak csoportja. Ebbe a csoportba tartoznak a lakossági fogyasztók, a kiefeszültségen vételező felhasználók, valamint a közfeladatot ellátó intézmények. Ezen fogyasztói szegmensbe tartozó fogyasztók Magyarország teljes áramfogyasztásának 26%-át teszi ki.[4]

2. Versenypiaci fogyasztók, egyetemes szolgáltatásra nem jogosultak. Ezen piaci szereplők piaci áron tudnak elektromos áramot vásárolni. A csoport résztvevői Magyarország teljes éves fogyasztásának 59%-át teszi ki.[4]

3. Erőművi villamosenergia felhasználók. Ebbe a speciális szegmensbe sorolható az elektromos áram előállítását végző termelési folyamatok energiaigénye, valamint a szállítás során keletkező energiaveszteség. Ezen fogyasztói klaszter Magyarország teljes fogyasztásának a 15%-át teszi ki.[4]



2. ábra: Magyarország áramfogyasztásának megoszlása a piaci szegmensek között[4]

### 1.1.5 Szabályozó hatóság

A szabályozó hatóság az energiapiac azon szereplője, amely összekapcsolja az előző négy csoportot és szabályozza azok működését. Mivel az átviteli hálózat egyetlen piaci szereplő kezében van, monopolhelyzettel bír a villamosenergia piacon. Szigorúan szabályozza a piac résztvevőit, mivel a rendszer megfelelő működése kiemelt fontosságú. Magyarországon ezt a szabályozást a Magyar Energia és Közműszabályozási Hatóság látja el.[2]

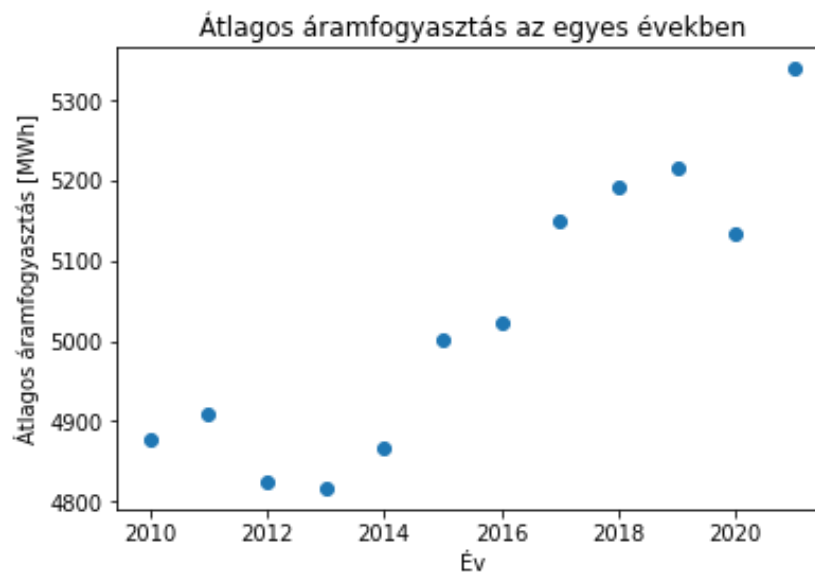
## 2 Magyarország áramfogyasztási adatainak elemzése

A legtöbb adatelemzési feladatnál az első lépés az adatok feltáró elemzése szokott lenni. Ekkor megvizsgálják a kiindulási adatokat és olyan összefüggéseket keresnek benne, amelynek segítségével hatékonyabbá tudják tenni a későbbi modellépítési feladatokat.

Munkám során én is egy előzetes feltáró adatelemzést végeztem a MAVIR Zrt. hálózatterhelési adatain. Ez az adathalmaz az elmúlt 10 év áramfogyasztási adatait tartalmazta MWh-ban megadva az idő függvényében 15 percenként aggregálva. Munkám során az egyes időtávokra vonatkozó átlagos rendszerterhelési adatokat vizsgáltam és összefüggéseket próbáltam keresni az adatok és a külső tényezők között. Az eredményeket pont-pont diagramok segítségével ábrázoltam. Az ábrák y tengelyének beosztása nem 0-ról indul, hogy jobban láthatóak legyenek az egyes időszakok közötti eltérések.[5]

### 2.1 Éves átlagos fogyasztás elemzése

Először azt vizsgáltam meg, hogy hogyan alakult az átlagos áramfogyasztás az elmúlt 10 évben. Ennek segítségével egyes időszakok trendjeit állapítottam meg, amelyekre próbálnak magyarázatot találni.



2.1. ábra: Magyarország átlagos áramfogyasztása az egyes években[5]

Azt tapasztaltam, hogy a 2010-es évek első felében 2010 s 2013 között kismértékben csökkent az éves átlagos fogyasztás az előző időszakhoz képest. Ez a 2008-as globális világgazdasági pénzügyi válsággal magyarázható, ugyanis ebben az időben Európában gazdasági visszaesés volt, ami a termelés csökkenésével járt.

Ez csökkenő trend 2014-től kezdve tört meg. Ezután egy hosszú növekedési szakasz kezdődött egészen a 2020-as évig. Ebben az időszakban egy globális gazdasági fellendülés ment végbe a világ minden országában, köztük Magyarországon is. Ezen évek során jelentős gyárépítések és termelési kapacitásbővítések történtek az országban. Ezen ipari létesítmények jelentősen járulnak hozzá az ország energiafogyasztásához, ezért érthető, hogy ebben az időszakban jelentősen megugrott az átlagos éves fogyasztás.

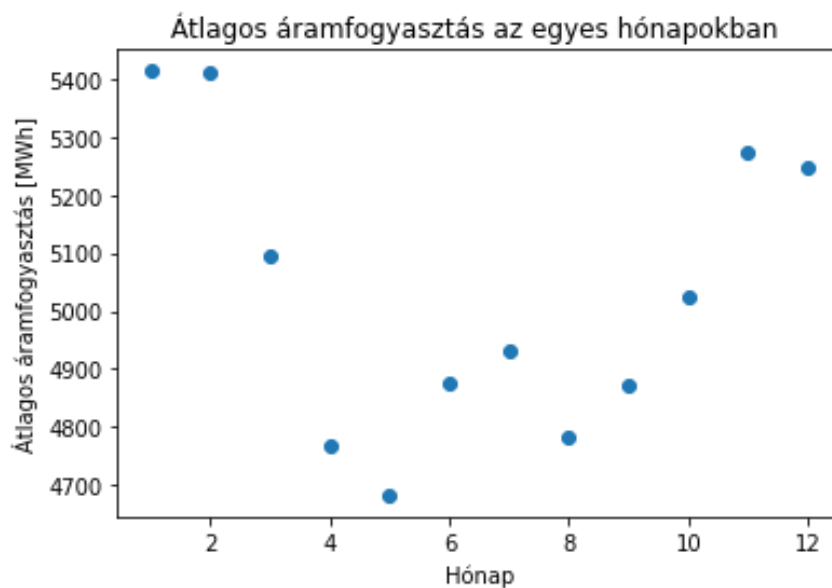
A 2020-as és 2021-es év első felében a világban egy globális koronavírus járvány tombolt, amelynek hatására jelentős ipari leállások és korlátozások voltak a világ minden országában köztük Magyarországon is. Ebben az időszakban a gazdasági termelés visszaesett és ezáltal az ipari létesítményekben is csökkent a termelés, ami magával vonzotta az átlagos áramfogyasztás csökkenését is.

Mivel Magyarországon az áramfelhasználás legnagyobb részét az ipari termelővállalatok fogyasztása adja, ezért erős kapcsolatot lehet feltételezni az éves fogyasztás és a gazdasági növekedés között. A fogyasztásra az áram világpiaci ára is hatással van. Ez főleg a 2022-es adatokon látszik. Ekkor az áramfogyasztás csökkent az előző évhez képest, míg a villamosenergia ára nagymértékben nőtt.

## **2.2 Havi átlagos fogyasztás elemzése**

Másodszor azt vizsgáltam meg, hogy milyen kapcsolat az havi átlagfogyasztás és a hónap sorszáma között. Azt tapasztaltam, hogy a havi átlagos rendszerterhelés nagymértékben függ a hónaptól.

Arra jutottam, hogy erős kapcsolat van az adott évszak és az átlagos fogyasztás között. A téli hónapokban sokkal nagyobb az átlagos áramfogyasztás, mint a nyári hónapokban. A tavaszi és őszi hónapok rendszerterhelési adatai egy csökkenő és egy növekvő trendet mutatnak. Azt is megállapítottam, hogy erős kapcsolat van a napi hőmérséklet és az átlagos áramfogyasztás értékei között. Ez leginkább a hűtési és fűtési folyamatok energiafelhasználásával magyarázható.



2.2. ábra: Magyarország átlagos áramfogyasztása az egyes hónapokban[5]

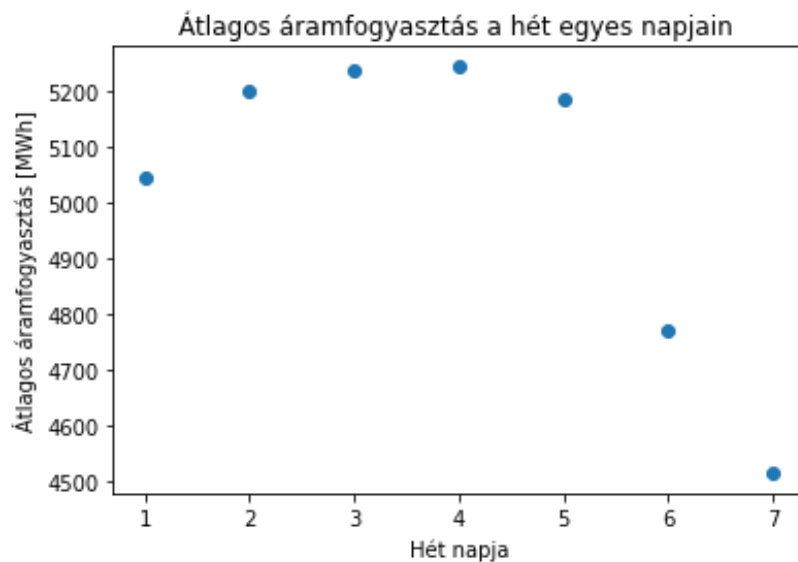
Az adatokból az olvasható ki, hogy a legnagyobb áramfogyasztás januárban, februárban, november és decemberben keletkezik. Ez azzal magyarázható, hogy ezek a leghidegebb hónapok és ezen hónapok során jelentős energiát használnak fel az épületek fűtésére. Ebben az időszakban az ipari üzemek fűtése adja az energiafelhasználás legnagyobb részét. A lakossági felhasználók is fűtenek ezen időszakban, de ez érdemben nem befolyásolja az országos áramfelhasználást, mivel a háztartási fogyasztók éves energiafelhasználása jóval kevesebb az éves ipari energiaigényeknél, valamint a lakossági fogyasztók esetében kevésbé elterjedt az elektromos árammal történő fűtés.

A tavaszi hónapok során egy csökkenő trendet figyelhetünk meg. Azt tapasztaljuk, hogy átlagos rendszerterhelés hónapról hónapra csökken. Ez a hőmérséklet folyamatos növekedésével magyarázható, mivel ilyenkor a fűtés energiaigénye csökken, mivel kisebb hőmérsékletváltozást kell elérni a fűtési folyamat során. Ennek az ellenkezőjét tapasztalhatjuk az őszi hónapokban, amikor a hőmérséklet folyamatosan csökken és hónapról hónapra nagyobb energiabefektetéssel jár a fűtés megvalósítása.

A nyári hónapokban ismét a fogyasztási adatok megugrását tapasztalhatjuk. Ebben az időszakban a hőmérséklet elég magas, amely túl magas az egyes ipari létesítmények megfelelő működése céljából. Ekkor hűteni szokták az ipari létesítményeket, ami szintén hozzájárul a fogyasztási adatok megugrásához.

## 2.3 Heti átlagos fogyasztás elemzése

A következő vizsgálati szempontom a napi átlagos fogyasztás elemzése volt. Megvizsgáltam, hogy milyen kapcsolat van a napi átlagos fogyasztás a nap héten belüli sorszáma között. Azt tapasztaltam, hogy hétköznap jóval nagyobb a fogyasztás, mint hétvégéken.



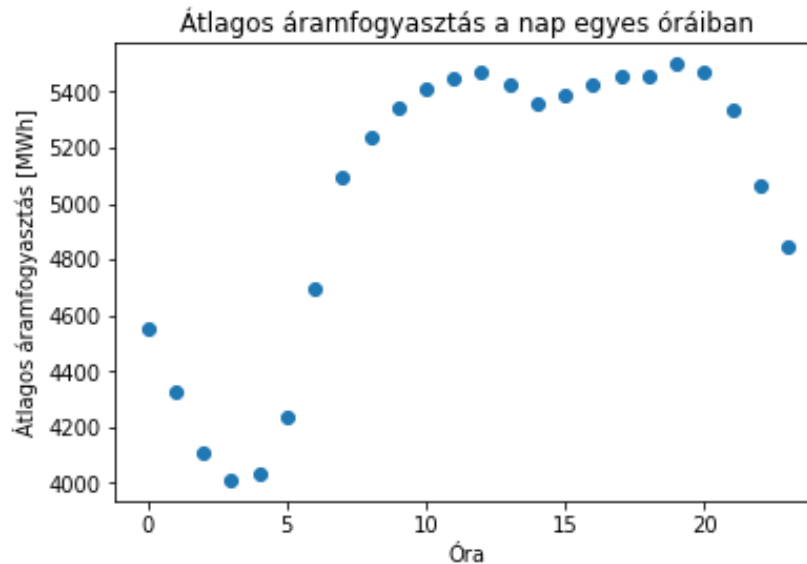
2.3. ábra: Magyarország átlagos áramfogyasztása a hét egyes napjain[5]

Az adatok alapján azt tapasztaljuk, hogy hétköznap jóval nagyobb a fogyasztás, mint a hétvégéken. A hétköznapoknál azt figyelhetjük meg, hogy a hétfői és pénteki nap fogyasztása elmarad a másik három hétköznapos nap fogyasztásától. Arra a megállapításra jutottam, hogy az adott nap munkarendi besorolása erősen befolyásolja az aznapi fogyasztást.

Ezen összefüggések a Magyarországi munkarenddel magyarázhatóak. A hétköznapos napokon a hivatalos munkanapokon jóval nagyobb a fogyasztás, mint a munkaszüneti hétvégi napokon. A legtöbb ipari létesítménybe csak hétköznap folyik termelés és hétvégén nem végeznek semmilyen tevékenységet. Persze vannak hétvégén is működő ipari egységek, de ezek száma elenyésző a többihez képest. A hétfői és pénteki napok fogyasztása azért lehet kevesebb a többi nap fogyasztásánál, mivel ezen a napokon már csak alacsonyabb határfokkal folyik érdemi munka. Pénteken már a nap vége felé leállítják a termelői egységeket, valamint gyakran ebben az időpontban végzik a karbantartásokat.

## 2.4 Napi átlagos fogyasztás elemzése

Az utolsó vizsgálatom arra irányult, hogy milyen összefüggés van az egy napi fogyasztási adatok között a nap egyes óráiban. Azt tapasztaltam, hogy a napon belüli fogyasztás leköveti az emberek valós munkarendjét.[5]



2.4. ábra: Magyarország átlagos áramfogyasztása a nap egyes óráiban[5]

Az éjszakai órákban tipikusan alacsonyabb a fogyasztás és ez az érték reggel 5 órától kezdve folyamatosan növekszik egészen 10 óráig. Ez azzal magyarázható, hogy reggel kezdődik el a hivatalos munka és szépen sorba ebben az időszakban indítják be a termelést az országban.

Dél előtt 10 óra és este 6 óra között a napon belüli fogyasztás általában egy stabil állandó értéket vesz fel, majd este 6 óra után kezd csökkenni újra. Ez azzal magyarázható, hogy ebben az időszakban van a fő munkaidő, ekkor üzemel a magyar ipar a teljes gyártókapacitáson.

Este 6 óra körül folyamatosan csökken a fogyasztás mértéke, mivel ekkor van vége a munkaidőnek és a termelési egységek leállnak vagy alacsonyabb termelési kapacitásra állnak át. A késő esti órákban pedig szintén csökken az áramfogyasztás mennyisége, mivel éjszaka a főbb fogyasztói tevékenységek leállnak és csak másnap reggel indulnak újra.



### 3 Idősoros adatok feldolgozásának módszerei

Idősoros adatnak nevezzük az olyan adathalmazokat, amelyben egy numerikus értéket az idő függvényében ábrázolnak. Ilyen idősoros adat például az áramfogyasztás az idő függvényében ábrázolva. Ezen adatok feldolgozása a klasszikus adatelemzési feladatokhoz képest nehezebb, mivel kezelni kell az idő dimenzió sajátosságait. Ezen adatok feldolgozására számtalan módszer ismert.

A legegyszerűbb módszerek az úgynevezett statisztikai módszerek. Ebbe a csoportba tartoznak a mozgóátlagszámításon alapuló, valamint ennek továbbfejlesztéseként az exponenciális-simítást alkalmazó módszerek, amik a mozgóátlagon kívül az idősoros adat egyes adatpontjaihoz egy fontossági súlyértéket is rendelnek. Ilyen modellek az ARIMA és SARIMA.[6]

Egy másik fő modellcsoportba a klasszikus gépi tanuláson alapuló modelleket soroljuk. Ekkor a statisztikai módszerek helyett gépi tanulást alkalmazunk, tehát különféle machine-learning algoritmusokkal a számítógépre bízunk, hogy megtalálja az összefüggéseket a múltbéli és jövőbéli adatok között és ezáltal képes legyen jó hatékonysággal előre jelezni a jövőt. Klasszikus módszerek közül döntési fákat is szoktak használni a feladat megoldására, valamint az ezen algoritmuson alapuló kombinált úgynevezett ensemble módszereket alkalmazzák. [6]

A harmadik csoportba a gépi tanulási módszerek egy speciális részhalmaza a deep-learning alapú módszerek tartoznak. Ekkor mesterséges neurális hálózatok segítségével próbáljuk megtanulni az adatokban rejlő összefüggéseket. Ezen módszerek nagy mennyiségű adat esetén bizonyulnak hatékonyak és az algoritmusok futásideje is jelentős. A deep-learning módszerek családjából az úgynevezett konvolúciós CNN neurális hálózatokat, valamint a memóriával rendelkező LSTM hálózatokat alkalmazzák. [6]

### 3.1 Gépi tanulási módszerek problémái

A gépi tanulás alkalmazása idősoros adatok feldolgozására felvet egy nagy problémát. Ez a probléma az adathalmaz dimenzióhiánya. Idősoros adatok feldolgozása során általában az adataink két dimenziósok. Az első dimenzió az idő dimenzió, míg egy második dimenzió egy numerikus dimenzió, ami az idő függvényében számszerű adatokat ábrázol. [6] [7]

A gépi-tanulási algoritmusok általában akkor működnek jól, amikor nagyobb az adathalmaz dimenziója. Különösen igaz ez a döntési fa alapú módszereknél, mivel ekkor a döntések egy-egy attribútum alapján történnek és több attribútum esetén alkotott döntési szabályok jobban reprezentálják a valós viszonyokat.

A problémát általában származtatott dimenziók generálásával oldják meg. Az idő függvényében ábrázolt értékek esetében gyakran meg lehet figyelni különféle trendeket és szezonális jelenségeket. Gyakran szezonálisak az adatok, tehát elkülöníthetünk különféle időszakokat, amelyek egy adott szabályosságot mutatnak. Ilyen időszak lehet például a hónap, hét, nap, óra.

A leggyakrabban ezen időszakok egymáshoz hasonlítanak, egy szabályos trendet követnek le. A legoptimálisabb esetben az adatok stabilitást mutatnak, tehát a várható értékük és szórásuk a teljes adathalmazban megegyezik. Ebben az esetben lehetne a leghatékonyabban rátanulni az adatok összefüggéseire, bár a gyakorlatban ritkán fordul elő, hogy az adatok ennyire szabályosak. [7]

Általában összegzik ezen időszakokat és különféle aggregátumfüggvényeket értékelnek ki az egy időszakra tartozó adatokon. A leggyakrabban alkalmazott aggregátumfüggvények az átlag, medián, minimum és maximum függvények.

Az aggregátumfüggvények által szolgáltatott eredményekből pedig új attribútumokat képeznek és felveszik ezeket az attribútumokat az adathalmazba. Ezzel növelik a dimenziószámot és segítségével a gépi tanulási modellek sokkal pontosabban meg fogják tudni tanulni az adatok közötti összefüggéseket. [7]

## 4 Gépi tanulási módszerek

Idősoros adatok előrejelzésekor általában a gépi tanulási modellek közül az úgynevezett regressziós modelleket használjuk. Ezen modellek segítségével egy folytonos értékészletű célváltozó értékét szeretnénk előre jelezni.

Ezen modellek alkalmazása esetén a célváltozó általában az idő függvényében ábrázolt numerikus érték, míg a bemeneti változók az idősoros adatok egyes időszakában az aggregátumfüggvényekkel előállított értékek szoktak lenni.

Az idősoros adatok előrejelzésénél az általában használt regressziós modellek nagyon érzékenyek a hiányzó adatokra, ezért nagyon fontos, hogy az adatokból nagyobb időintervallumok ne maradjanak ki. Általában törekedni kell folytonos kimaradásmentes adatok használatára.

Egy másik nagyon fontos feladat annak biztosítása, hogy az adatok mintavételezése állandó legyen a teljes adathalmaz esetében, tehát az időtengelyen bármely két adatpont között azonos konstans időnek kell eltelnie.

### 4.1 Gépi tanulási modellek hatékonyságának mérése

A regressziós gépi tanulási modellek hatékonyságát különféle hibametrikákkal mérjük. A modell kiértékelése során egy adott tesztalmaz esetében összehasonlítjuk a modell által prediktált értéket összehasonlítjuk a valós értékekkel. A modellünk annál jobb, minél közelebb van a modell által jóslt érték a valós értékhez.

A leggyakrabban az alábbi hibametrikákat alkalmazzuk:

- Átlagos hiba (ME – Mean Error)
- Átlagos négyzetes hiba (MSE – Mean Squared Error)
- Átlagos abszolút hiba (MAE – Mean Absolute Error)
- Átlagos százalékos hiba (MAPE – Mean Absolute Percentage Error)

A gépi tanulási modellek építése során mindig egyfajta hibametrikára optimalizáljuk a modell működését. Amennyiben helyesen modellezünk a többi hibametrikának is viszonylag hasonlóan kell viselkednie, mint a kiválasztott optimalizációs metrika. [11]

A hibametrikák az alábbi képletek segítségével számíthatóak ki, ahol  $n$  jelöli a validációs tesztadathalmaz méretét,  $y_{real}$  jelöli a célváltozó valós értékét, míg  $y_{predicted}$  a regressziós modell által meghatározott célváltozó értéket jelöli.

$$ME = \frac{1}{n} \sum_{i=1}^n (y_{real_i} - y_{predicted_i})$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_{real_i} - y_{predicted_i})^2$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_{real_i} - y_{predicted_i}|$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_{real_i} - y_{predicted_i}}{y_{real_i}} \right|$$

4.1. ábra: Hibametrikák kiszámítása [11]

## 4.2 Döntési fák

A döntési fák az egyik legegyszerűbb és leggyorsabb gépi tanulási algoritmus, ami még a mai napig is a gyakorlatban széleskörűen használt, valamint erre épülnek a közkezdvelt összetett modellek is.

A regressziós döntési fák működése azon az alapgondolaton alapszik, hogy minden regressziós feladat egyszerű döntési feladatokra vezethető vissza. Egy ismeretlen adatminta feldolgozásakor a fa gyökeréből indulunk ki és a bemenő attribútumokra vonatkozó értékek függvényében különféle szabályok alapján lépünk a fában. Az egyes csomópontokban döntéseket hozunk, az adatminta adatai alapján, ezen döntések egyértelműek, vagyis minden egyes csomópontban egyértelműen eldönthető, hogy melyik csomópontra kell ugrani a fában. Addig folytatjuk a lépkedést a fában, amíg egy levélelembe nem érünk, ahonnan már nem tudunk továbbmenni. Ekkor a modell által kimenetként adott érték a levélelemhez rendelt becsült érték lesz. [8]

Egy regressziós döntési fát egy rekurzív algoritmussal vagyunk képesek előállítani egy tanító adathalmaz felhasználásával. Az algoritmus kezdetkor a teljes tanító adathalmazból indulunk ki, majd ezt az adathalmazt próbáljuk két részre vágni valamely bemeneti attribútum értéke alapján úgy, hogy a keletkező két darab újonnan létrehozott adathalmazra teljesüljön, hogy a célváltozó értéke kevésbé szórjon a vágás

után, mint azelőtt. A vágási feltételt rendeljük hozzá a gyökércsomóponthoz, és ennek a csomópontnak a két gyerekéhez rendeljük hozzá a két újonnan létrehozott adathalmazt. Ezt az eljárást rekurzívan futtatjuk le a keletkező csomópontra és ezt addig folytatjuk, amíg nem teljesül, hogy a fa leveleihez tartozó adathalmazokban a célváltozó szórása egy adott kis értékre nem csökken. Ekkor elkészült a döntési fánk, a levélelemekhez ekkor hozzárendeljük a csomópontokhoz tartozó adathalmaz célváltozóinak átlagát. Az egyes csomópontokhoz rendelt vágási szabályok alapján döntünk a fa végrehajtása során, amíg el nem jutunk egy levélelemig. Ekkor a fa által prediktált érték a levélelem címkéje lesz. [9]

A fa felépítése során figyelni kell arra, hogy ne épüljön túl mély fa, mivel a nagy mélységű fák felépítése több időt vesz igénybe. Szintén ekkor megvan a veszélye annak is, hogy egy tanítóadathalmazra optimalizált nagyon specifikus túltanult fa jön létre. Egy ilyen túltanult döntési fa kevésbé pontos predikciókat ad egy másik hasonló adathalmazra. Ezért mélységkorlátozni szokták a fákat, úgy, hogy amennyiben a fa felépítése során elértünk egy adott mélységet, akkor az algoritmust azon a ponton leállítjuk. Az épülő döntési fa maximális mélysége az egyik legfontosabb paramétere a döntési fa alapú modelleknek. [9]

A regressziós döntési fával egy hatékony eszközt kapunk a kezünkbe prediktív analitikai feladatok megoldásához. Viszonylag gyors eljárásról beszélünk, tehát viszonylag alacsony futásidővel képesek vagyunk a modelleket betanítani. Egy másik nagy előnye ennek a megoldásnak, hogy segítségével felderíthető, hogy egy adathalmazban mely bemeneti változók befolyásolják legjobban a célváltozó értékét. Mivel döntések segítségével jutunk el a fa gyökerétől egy levélelemig, ezért a döntés folyamata interpretálható, és a modell által megtalált összefüggésekből kinyert tudás felhasználható emberi tanulás és vállalati folyamatoptimalizáció céljára. [9]

### **4.3 Ensemble módszerek**

Az ensemble módszerek lényege, hogy nem egyetlenegy modellt használunk a regressziós probléma megoldásához, hanem sok különálló modellt építünk fel és együttesen alkalmazzuk azokat a feladat megoldása során. Az ensemble módszerek egyik legfontosabb paramétere az almodellek száma, ami azt fejezi ki, hogy hány különálló modell kerül felépítésre a végeredmény meghatározásához. Kimenatként általában, de

nem mindig a sok különálló modell által meghatározott kimenetek súlyozott átlagát számítjuk ki, és ez lesz a végeredmény.

Ezen módszerek használatával sokkal kisebb hibával tudjuk megoldani a regressziós problémákat, mintha csak egyetlenegy modellt használtunk volna a predikció végrehajtásához. Az ensemble módszerek három fő családba sorolhatóak, megkülönböztetünk bagging, stacking és boosting módszereket. [8]

Az első csoportot a bagging módszerek alkotják. Ezen eljárások esetében több modellt építünk fel a tanító adathalmaz alapján úgy, hogy az egyes modellek a tanító adathalmaz bemeneti változóinak más és más részhalmazát kapják meg bemenetként. A módszer esetében az egyes modellek típusa és paraméterezése általában megegyezik, de az egyes almodellek tanító adathalmaza eltérő, mivel nem tartalmazzák az összes bemeneti változó értékét, hanem annak csak egy kisebb részhalmazát. Az egyik legismertebb ilyen bagging algoritmus a döntési fákon alapuló RandomForest algoritmus.

A második csoportba tartozó stacking módszerek nagyon hasonlóak a bagging módszerekhez, de ebben az esetben nem azonos típusú és paraméterű modelleket használunk fel, hanem a felépített különálló modellek különböznek egymástól. Ebben az esetben nem feltétlenül történik vágás a bemeneti attribútumok esetében sem, gyakran a teljes adathalmaz minden bemeneti paramétereit felhasználják egy ilyen modell felépítése során. [10]

A harmadik csoportot a boosting módszerek alkotják. A boosting módszerek esetében általában a felépülő különálló modellek mind a típusaikban, mind paramétereikben és a felhasznált attribútumok esetében is megegyeznek. Ezen modellek közötti különbséget az adja, hogy a kiindulási tanító adathalmaz elemei más-más súllyal szerepelnek az egyes modellek tanító adataiban. Az első modell építésénél felhasználjuk a teljes kiindulási tanító adathalmazt, majd megvizsgáljuk, hogy az így felépülő modell mekkora pontossággal határozza meg a célváltozó értékét a tanító adathalmaz esetében. Ezután a következő modell esetében a tanítóhalmazt úgy állítjuk elő, hogy az előző modell által viszonylag kis hibával meghatározott elemek súlyát csökkentjük, míg nagy hibával meghatározott elemek súlyát növeljük a következő modell tanító adathalmazában. Ezt az eljárást addig folytatjuk, amíg fel nem építettük a megfelelő számú modellt. Az egyes modelleket ezután aszerint súlyozzuk, hogy mennyire jó eredményt adtak a tanító adathalmaz esetében. Ezen modellek eredményeinek a súlyozott átlaga fogja adni a végeredményt. Az egyik ismertebb boosting algoritmus a GradientBoosting algoritmus.

## 5 Az implementáció során felhasznált technológiák

Ebben a fejezetben bemutatom a megoldás implementációja során használt főbb technológiákat, platformokat és könyvtárakat. Röviden ismertetem azok főbb jellemzőit és használati eseteit.

### 5.1 A Python nyelv

A Python programozási nyelv a manapság leggyakrabban használt programozási nyelvek közé tartozik. Egy általános célú, nagyon magas szintű nyelv, melyre általában egyszerűsége, könnyen tanulhatósága, valamint hatékonysága miatt esik a választás a fejlesztés során.

A nyelvet Guido van Rossum kezdte el fejleszteni a 1980-as években. A mostani legfrissebb főverziója a 3-as verzió, ami 2008-ban került kiadásra. Tervezése során a futási sebesség helyett a kód olvashatóságára és a programozói munka megkönnyítésére helyezték a hangsúlyt.

A Python nyelv egy úgynevezett interpretált nyelv. Az elkészített Python forráskódot egy értelmező segítségével azonnal futtatni lehet, nincs szükség a forráskód előzetes lefordítására. Script nyelvnek is szokták tekinteni. Teljesítményben elmarad a C nyelvtől, de léteznek nagyon hatékony Python könyvtárak, amelyek C nyelvhez hasonló teljesítményt nyújtanak. Ezeket a könyvtárakat általában nagy számításigényű tudományos és adatkezelési műveletekre használják.

A nyelv számtalan programozási paradigmát támogat. Segítségével fejleszthetünk objektum-orientált, procedurális, strukturális és funkcionális szemléletű programokat. Dinamikus típusos nyelv, tehát az egyes adatobjektumok típusát nem szükséges futásidőben definiálni, és ez a típus megváltozhat futás közben. Ennek ellenére egy szigorú típusrendszerrel rendelkezik.

A nyelv használata mellett egy másik nagy érv, hogy automatikus memóriakezelést valósít meg. Ez megkönnyíti a fejlesztők munkáját, mivel nem kell foglalkozniuk a memóriakezeléssel. A memóriakezelés a Java és .NET platformokhoz hasonlóan szemétyűjtő komponenst használ, ami törli a memóriából azon adatobjektumokat, amelyekre nem mutat referencia egy adott pillanatban.

## 5.2 A Numpy könyvtár

A Numpy könyvtár a Python környezet egyik legismertebb könyvtára. Ezt a könyvtárat leggyakrabban tudományos matematika számítások elvégzésére használják.

A könyvtár egyik fő előnye, hogy natív szintű gyorsasággal képes matematikai számításokat, főként mátrixműveleteket elvégezni. Ezért leggyakrabban lineáris algebrai számítások elvégzésére szokták használni. Ez a könyvtár gyorsan végzi el a vektor és mátrix műveleteket, de viszonylag lassan számolja ki az ettől eltérő műveleteket, ezért igyekezni kell mindent lineáris algebrai műveletek segítségével leírni.

Ez a könyvtár alapja a főbb tudományos Python könyvtáraknak, valamint a gépi tanulást alkalmazó eszközöknek is. Ezen könyvtár API-ját használják a legismertebb gépi tanulási keretrendszerek a Scikit-learn, Keras, Tensorflow és a Pytorch könyvtárak, amelyek hatékonyságához elengedhetetlen, hogy gyorsan végrehajthatóak legyenek a mátrixműveletek.

## 5.3 A Pandas könyvtár

A Pandas a Python környezetben végzett adatelőkészítések során leggyakrabban használt osztálykönyvtár. Segítségével könnyedén és hatékonyan lehet az adatok előfeldolgozását megvalósítani. Ez a könyvtár a Numpy könyvtárra épül, tehát gyors a végrehajtás, így nagyméretű adatok esetén is viszonylag gyorsan vagyunk képesek az adatfeldolgozási lépéseket megvalósítani.

Definiál egy speciális adatszerkezetet, a DataFrame-t, amely segítségével hatékonyan kezelhetőek nagyméretű táblázatos adatok. Lehetőségünk van különféle adatforrásokból adatokat beolvasni, valamint számtalan formátumba tudjuk kiexportálni az előkészített adatainkat is.

Számos adatfeldolgozási művelet áll rendelkezésünkre, az adattáblákat különféle szempontok alapján tudjuk szűrni, valamint hatékonyan módosítani is tudjuk az elemeit. A különféle adattáblákat hatékonyan tudjuk illeszteni egymáshoz, és ezáltal több különálló adatforrást egyesíteni.

Lehetőségünk van aggregálni az adatokat és a leggyakrabban használt aggregátumfüggvényeket is futtatni tudjuk az adatokon. Hatékonyan tudunk minimum, maximum, átlag, és medián értéket számolni.



## 5.4 A Scikit-Learn könyvtár

A Scikit-Learn könyvtár a Python programozási környezet egyik legismertebb és a legtöbbször által használt gépi tanulási könyvtára. Ez a könyvtár a Numpy alapokra építve hatékony machine-learning modellfejlesztést tesz lehetővé. A könyvtár a fejlesztési folyamat teljes spektrumát lefedi, a folyamat minden lépése számára biztosít megoldásokat.

Számos adatelőkészítési lépés áll rendelkezésre, amely a Numpy és a Pandas könyvtárra épít. Lehetőség nyílik az adatok skálázására, standardizálására, valamint a kategorikus változók különféle numerikus kódolására is. Ezen felül lehetőség van különféle dimenziócsökkentési numerikus módszerek alkalmazására és a hiányzó értékek meghatározására is.

A keretrendszer jelentős számú gépi tanulási modell típust támogat. Lehetőség nyílik felügyelt és nem felügyelt tanulás megvalósítására is. Támogatottak a különféle távolság és sűrűség alapú klaszterezési algoritmusok. A főbb felügyelt tanulási, osztályozási és regressziós modellek széles köre van implementálva a keretrendszerben. Elérhetőek a lineáris regressziós modellek, a logisztikus regressziós modellek, a K legközelebbi szomszéd módszert alkalmazó modellek, a döntési fák, a naive bayes modellek, valamint a support vektor gépek alkalmazására is lehetőség nyílik. Megtalálhatóak a neurális hálózat alapú modellek is, valamint a több modellt összefogó úgynevezett ensemble alapú módszerek is.

A modellek kiértékeléséhez számtalan metrika kiszámítása implementálva van ebben az osztálykönyvtárban. Segítségével a legtöbb kiértékelési metrika könnyen és gyorsan kiszámolható a megalkotott modellek teljesítménymérése céljából. A kiértékeléshez, a különféle tanító és tesztadathalmaz generáláshoz, keresztvalidációhoz használható funkciók is implementálva vannak.

A modellek értelmezéséhez és a feature-engineering feladathoz is számos megoldás áll rendelkezésre. A modellek optimalizálásához számos automatikus hiperparaméter optimalizációs eljárás is megvalósításra került ebben a szoftverkönyvtárban.

## 5.5 A Dateutil könyvtár

A Dateutil könyvtár a Python környezet egyik közkedvelt segédkönyvtára, amely segítségével idősoros adatokat tudunk könnyedén feldolgozni. Ezen segédmodullal a dátumkezelési műveletek gyorsan elvégezhetők

A könyvtárral egyszerűen tudunk például dátum és idő, valamint időintervallum közötti műveleteket elvégezni, amely nagyban megkönnyíti például az idősoros adatok feldolgozását, a különféle időablakos mintavételezési feladatok kivitelezését, segítségével adott időszakra vonatkozó adatagregálási feladatok is hatékonyan hajthatók végre.

## 5.6 A Matplotlib könyvtár

A Matplotlib könyvtár a Python környezet első számú és a leggyakrabban használt adatvizualizációs könyvtára. A segítségével statikus és dinamikus interaktív adatvizualizációs ábrák készíthetők.

Ez a vizualizációs környezet számos diagramtípust támogat. Segítségével képesek vagyunk hisztogramokat, oszlop vagy sordiagramokat, tortadiagrammokat, pontdiagramokat, boxplotokat, vonaldiagramokat készíteni. Lehetőségünk van több alapdiagramból felépített összetett vizualizációs objektumokat is felépíteni.

Az adatok vizualizációja mellett néhány egyszerűbb adattranszformációra is képes. Különböző médiatetartalmak, például képek vizualizációját is támogatja.

Az ebben a környezetben készült ábrák jól együttműködnek a Jupyter notebook fejlesztési környezettel és egyszerűen jelennek meg a notebookokban. Gyakran alkalmazzák gépi tanulási modellek fejlesztése során az adatok és az eredmények vizualizálása céljából.

## 6 Áramfogyasztás előre jelző modellek bemutatása

Munkám során az országos villamosenergia-hálózat rendszerterhelésének idősoros előrejelzésével foglalkoztam. A becslést mindig egy adott bázisdátumtól viszonyított következő N órára végeztem el. Az általam elkészített algoritmus tehát két fő paraméterrel rendelkezik. Ez a két paraméter a bázisdátum és az N numerikus paraméter.

A modellek építéséhez a MAVIR Zrt. 2011-2021 közötti áramfogyasztási adatait használtam fel 15 perces mintavételezéssel. A modell tanításánál a kezdeti dátumtól a bázisdátumig tartó időszakok adatai segítségével tanítottam be a modellt és a következő N óra adatai alapján teszteltem a hatékonyságát.

A modellezés során a modelleket a következő 48 órára történő predikció esetére optimalizáltam. Az algoritmusom paraméterezhető, de különféle időtávokra más modellek adják a legjobb eredményt. Nem tudtam olyan modellt készíteni, ami N minden értéke esetén a legjobb eredményt adja. Azért döntöttem a következő 48 óra mellett, mivel a valós ipari gyakorlatban is általában ezt használják. Az ipari predikciós rendszerek minden reggel futtatják a modelleket és a következő 1-2 napot jelzik előre, amely adatok segítségével tudják a következő napok szabályozását a rendszerszabályozók megtervezni.

A modellek építése során az átlagos százalékos hibára optimalizáltam, mivel az energetikai folyamatokat a valós gyakorlatban általában ezen metrikával jellemzik. Minden általam felépített modell bemutatásakor a többi metrikát is megadom a modelljeim esetében, de az optimalizálás során a százalékos hibát vettem alapul.

A tútanulás ellen úgy védekeztem, hogy a modell elkészítésénél 5 különböző tesztalmazon ellenőriztem a modellem hatékonyságát és figyeltem arra, hogy ne legyen túl nagy eltérés az egyes tesztalmazokon mért hibametrikák értékei között. A kiértékeléskor még így is volt néhány időszak, amely pontossága jelentősen eltért az átlagostól, de ezek száma nem volt túl nagy.

A modellépítési stratégiát a folyamat végén úgy értékeltem ki, hogy az elmúlt 5 év adataiból véletlenszerűen választottam ki minden évből 5 dátumot. Ezen dátumokat követő 48 óra fogyasztását próbáltam megbecsülni és ez alapján értékeltem ki a modellek teljesítményét.

## 6.1 Mavir Zrt. adathalmazának előfeldolgozása

Az adathalmazt különféle időszakok mentén aggregáltam és az egy időszakba tartozó adatokra különféle aggregátumfüggvényeket futtattam. Ezen aggregátumfüggvények értékei szolgáltatták az egyes mesterségesen generált oszlopok értékeit. Az adatokat évek, hónapok, napok és órák szerint aggregáltam és minden ilyen időszakra meghatároztam az időszakba eső fogyasztási adatok minimum, maximum, átlag és medián értékét.

Az eredeti adathalmazhoz hozzárendeltem ezen időszakok aggregált értékeit. A hozzárendelésnél figyeltem arra, hogy a bázisdátumnál későbbi adatok alapján számolt aggregátumértékek ne kerülhessenek a tanító adathalmazba. Például, ha a következő 48 órára készítjük az előrejelzést, akkor egy céldátumot megelőző nap aggregátumait nem használhatjuk fel, csupán csak a kettővel megelőző nap adatait.

A fogyasztás másképpen alakul az év egyes hónapjaiban, valamint a hét egyes napjain is. A fogyasztás viszonylag egy szabályos trendet követ. Ezért az adathalmaz dátumai alapján label-encode technikával külön oszlopokat generáltam, amely segítségével reprezentáltam, hogy az adott adatpont melyik hónapra és napra esik, hogy ezzel is segítsem a gépi tanulási algoritmusok tanulási folyamatát.

Az adathalmazt ezután két részre vágtam: egy tanuló és egy teszt halmazra. A tanuló halmazt a kezdeti adatoktól egészen a bázisdátumig terjedő időszak adatai alkotják, ezt használtam fel az általam megalkotott gépi tanulási modellek feltanítására. A teszthalmazt a bázisdátumot követő  $N$  óra adatpontjai alkotják, ezek segítségével értékeltem a modellek teljesítményét.

## 6.2 Bemeneti változók fontosságának vizsgálata

Az adatelemzési feladatok egyik fontos lépése az úgynevezett feature-engineering. Ez magában foglalja azt, hogy az adathalmazunk változói közül kiválasszuk a legfontosabb változókat, amelyek legjobban befolyásolják a kimenet eredményét. Általában a modellünk hatékonysága nagyobb lesz, ha csak a valóban fontos adatokat használjuk fel a tanítás során, az érdektelen nem befolyásoló adatok rontják a tanítás teljesítményét. [12]

A folyamat végrehajtására számos módszer ismert. Munkám során kihasználtam, hogy a döntési fán alapuló gépi tanulási modellek különféle feltételek alapján vágják darabokra az adathalmazt, és ez alapján döntési feltételeket készítenek. Ezen modellekből kinyerhető, hogy az egyes változókat mennyire tartja a modell fontosnak a kimenet szempontjából.

Egy GradientBoosting regressziós modellt tanítottam fel a teljes adathalmazon, hogy megvizsgáljam, mely változók határozzák meg legnagyobb mértékben a kimenetet. A modelltől kinyertem a változók fontosságát és ez alapján döntöttem, hogy melyeket fogom felhasználni a tanítás során. [12]

A legnagyobb mértékben az elmúlt N óra időszakának és az elmúlt nap időszakának átlagos, medián, minimum és maximum értékei járultak hozzá a kimenethez. Kisebb mértékben ugyan, de még az elmúlt hónap és év aggregált értékei is hangsúlyosak.

A label-encode-olt hónapok és napok közül is voltak olyanok, amelyek jelentősen befolyásolták a kimenetet, míg mások szinte alig. A hónapok közül az április, a november és a decembert reprezentáló oszlop értékeit használta fel jelentősebben a modell a predikció során, míg például a február, június, július és szeptember hónapot jelző mezők értékét szinte teljesen használhatatlannak ítélte. [12]

## 6.3 Az alapmodell bemutatása

A legalapvetőbb kezdeti modell során csak a MAVIR Zrt. nyilvános adatait használtam fel. Ezen kezdetleges modell esetén nem vettem figyelembe se az időjárást, sem pedig az ország munkarendjét az előrejelzés során.

A modell bemeneti paramétereként felhasználtam az elmúlt N óra, nap, hónap és év átlagos, medián, minimum és maximum fogyasztási értékeit. Ezen kívül még bemeneti változóként a fogyasztást legdominánsabban befolyásoló hónapok és napok one-hot-encode-olt oszlopait használtam fel. Ezek a június, augusztus, szeptember és december hónapok voltak, míg a napok esetében a kedd, szerda, csütörtök, péntek és vasárnap.

A modell felépítéséhez az ensemble módszerek családjába tartozó RandomForest algoritmus regressziós változatát használtam fel. A modell hiperparamétereként megadtam, hogy a modell hány döntési fát foglaljon magában, valamint korlátoztam a modell döntési fáinak maximális mélységét is, hogy ne tanuljon túl. Ezen kívül a random generátor kezdeti értékét is fixáltam, hogy a modell tanítása determinisztikus legyen.

Hiperparaméter neve	Értéke
Belső modellek száma(n_estimators)	700
Maximális mélység(max_depth)	20
Random generátor érték(random_state)	42

6.1. táblázat: A modell hiperparaméterei

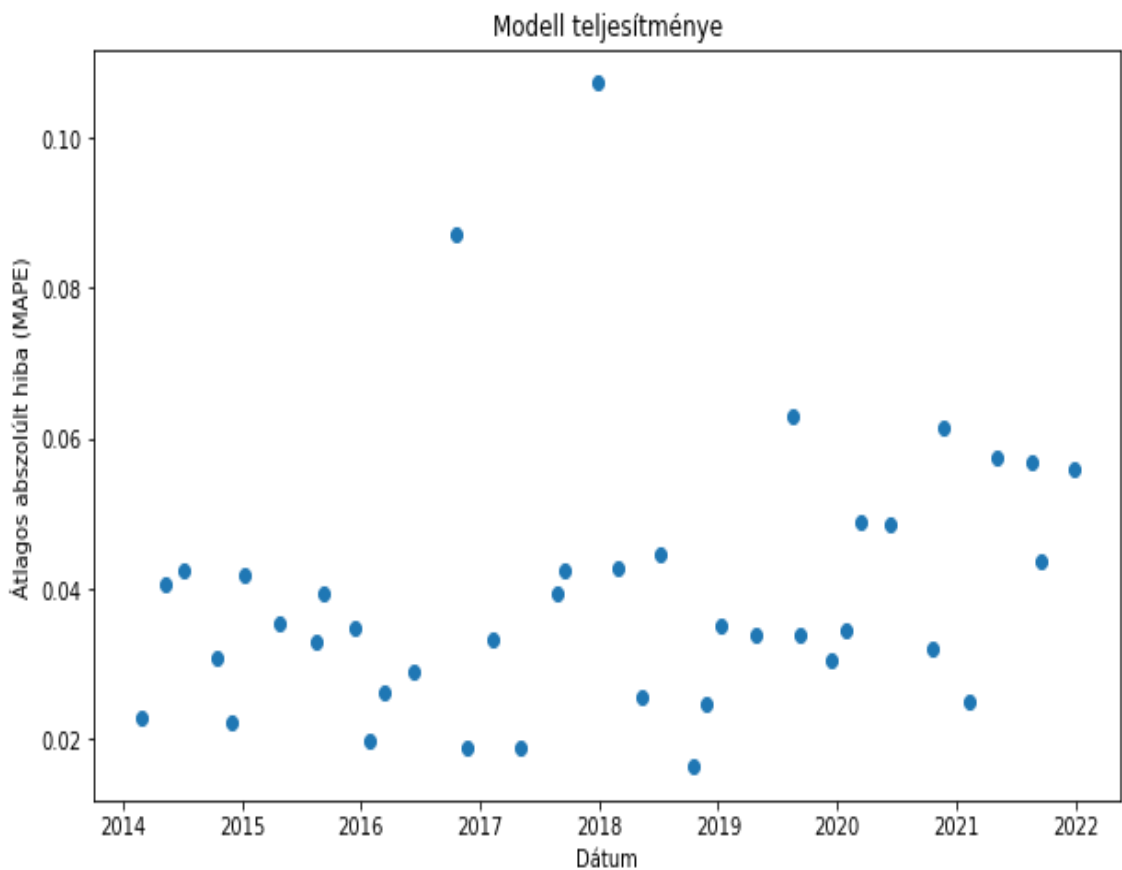
A modell teljesítményét 35 különböző bázisdátum és N=48 paraméter esetén teszteltem. Azt tapasztaltam, hogy az adatok többsége esetében átlagosan 2%-6% közötti átlagos abszolút hibával képes előre jelezni az áramfogyasztást, de van egy-két szélsőséges eset, amely esetében rossz pontosságot szolgáltat. Utóbbi adatpontok általában nagyobb ünnepnapok közelében helyezkednek el, és jelentősen eltérnek a többitől, ezért ezekre a napokra a modell nem volt képes rátanulni.

A modell teljesítményét különféle metrikák alapján kiértékeltem és meghatároztam a hibaszámok, minimum, maximum, átlag és medián értékét.

	Átlag	Minimum	Maximum	Medián
<b>MSE</b>	81602,183367	12732,973989	428419,577461	60205,695646
<b>MAE</b>	203,605977	87,548716	500,689623	188,710585
<b>MAPE</b>	0,039535	0,016469	0,107233	0,034932

6.2. táblázat: A modell hibametrikái

A modell teszhalmazon történő kiértékelési eredményeit egy pontdiagrammon ábrázoltam, amely a tesztbázisdátumok függvényében mutatja a modell átlagos abszolút százalékos hibáját.



6.1. ábra: A modell átlagos abszolút százalékos hibája a tesztadatokon

## 6.4 Hőmérsékletet alkalmazó modell

Ahogy korábban is megállapítottam az áramfogyasztást jelentősen befolyásolja az aktuális környezeti hőmérséklet, ezért a pontosabb előrejelzés céljából a modell bemeneteként hőmérséklet- adatokat is felhasználtam.

Mivel általában napjainkban a hőmérséklet előrejelzése precízen még nem megoldható, ezért én sem használtam fel valós hőmérséklet adatokat, hanem minden adatponthoz az azt megelőző N órával ezelőtti hőmérsékleti értéket rendeltem hozzá.

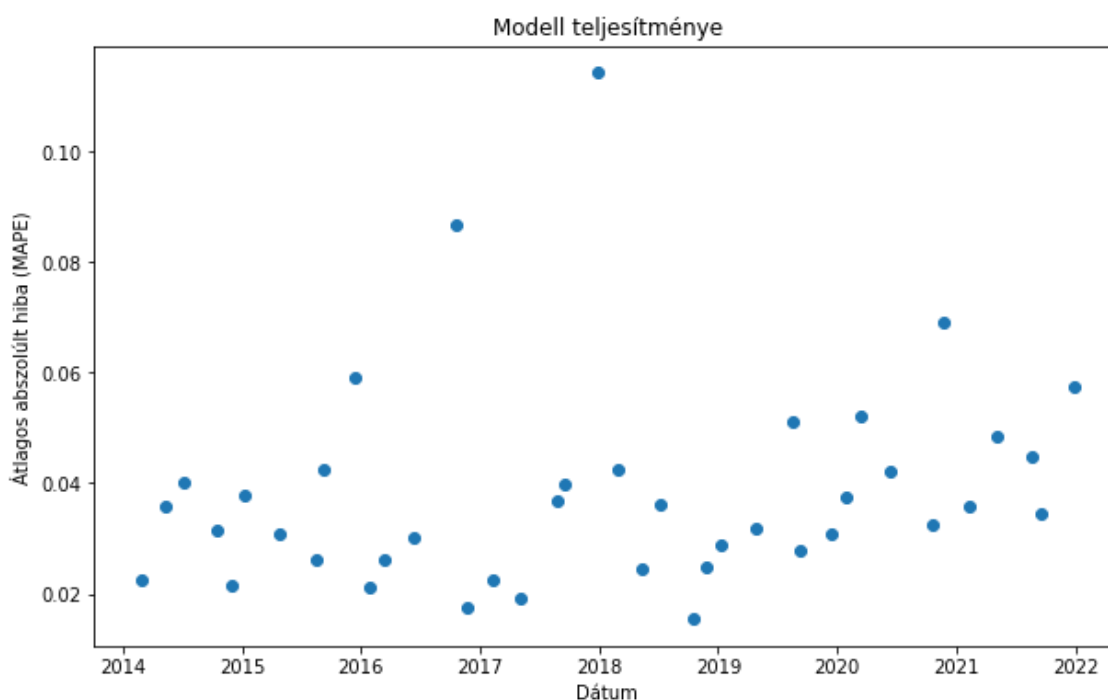
Az általam felhasznált adatokat az OpenWeatherMap nemzetközi időjárás adatszolgáltatótól szereztem be, amely az óránkénti magyarországi átlag hőmérsékleteket tartalmazta az elmúlt 10 évre visszamenőleg. Ezen adathalmaz adataival bővítettem a MAVIR adatokat tartalmazó adathalmazt, majd egy RandomForest modellt tanítottam fel a korábbi modellel azonos paraméterekkel.

A modell teljesítményét a korábban ismertetett 35 teszt dátumot követő 48 órányi fogyasztás predikciójával teszteltem. A modell teljesítménye kis mértékben átlagosan kb. 0,1%-kal javult a korábbihoz képest. Tehát nagy javulást nem hozott a hőmérsékleti adatok felhasználása, a korábbi modell által gyengén előrejelzett időszakokat ezt a modell sem tudta jobban előre jelezni.

	<b>Átlag</b>	<b>Minimum</b>	<b>Maximum</b>	<b>Medián</b>
<b>MSE</b>	81553,890579	11400,978057	442187,240044	51048,251797
<b>MAE</b>	198,901425	82,127386	535,828990	170,02023
<b>MAPE</b>	0,038298	0,015541	0,114230	0,035054

6.3. táblázat: A modell hibametrikái N órával késleltetett hőmérsékleti adatok esetén



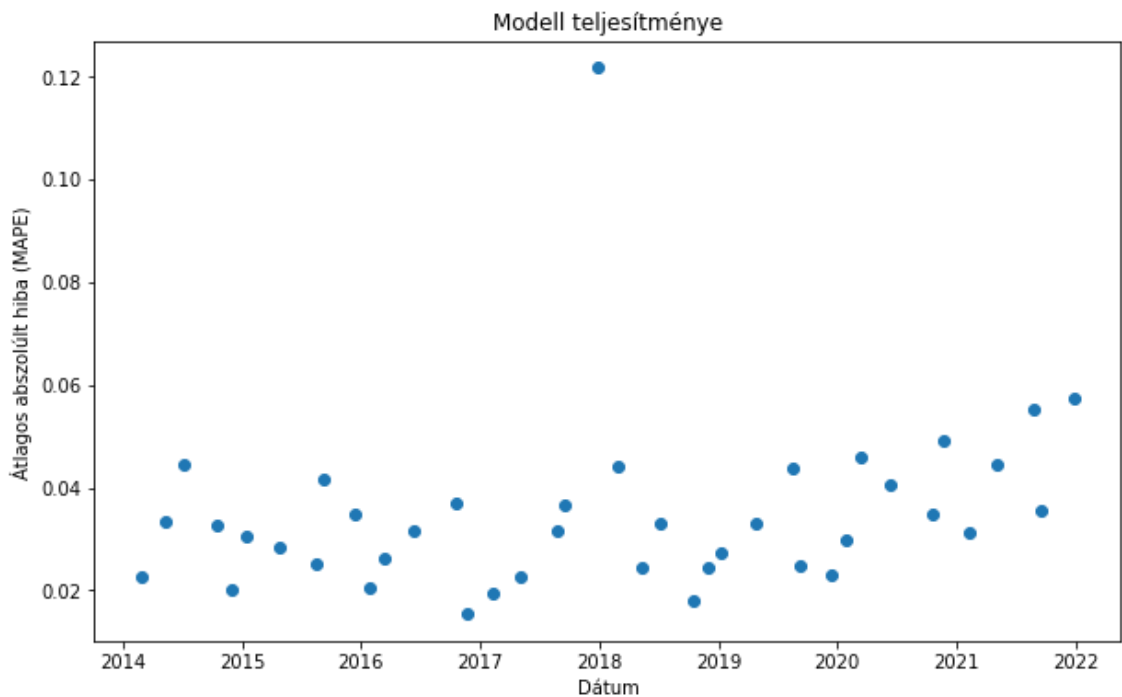


**6.2. ábra: A modell átlagos abszolút százalékos hibája N órával késleltetett hőmérsékleti adatok felhasználásával**

Érdekességként azt is megvizsgáltam, hogy mekkora lenne a modell teljesítménye, ha a valós hőmérsékleti adatokat használnánk fel a késleltetett adatok helyett. Ekkor átlagosan 0,35%-kal javult a modell teljesítménye az alapmodellhez képest. Ez a megoldás akkor lenne releváns, ha rendelkeznénk egy 100%-os pontosságú hőmérséklet előrejelzővel.

	<b>Átlag</b>	<b>Minimum</b>	<b>Maximum</b>	<b>Medián</b>
<b>MSE</b>	81553,890579	11400,978057	442187,240044	51048,251797
<b>MAE</b>	198,901425	82,127386	535,828990	170,02023
<b>MAPE</b>	0,038298	0,015541	0,114230	0,035054

**6.4. táblázat: A modell hibametriái valós hőmérsékleti adatok használata esetén**



6.3. ábra: A modell átlagos abszolút százalékos hibája valós hőmérsékleti adatok felhasználásával

## 6.5 Munkarendet alkalmazó modell

A hőmérséklet mellett a másik fontos áramfogyasztást befolyásoló tényező az ország munkarendje. Ezen modell esetében az alapmodellt úgy fejlesztettem tovább, hogy munkarendet is felhasználtam bemeneti adatként. Egy munkanap oszlopot generáltam az adathalmazban, amely értéke 1, ha az adott nap munkanap és értéke 0, ha a megadott nap munkaszüneti nap.

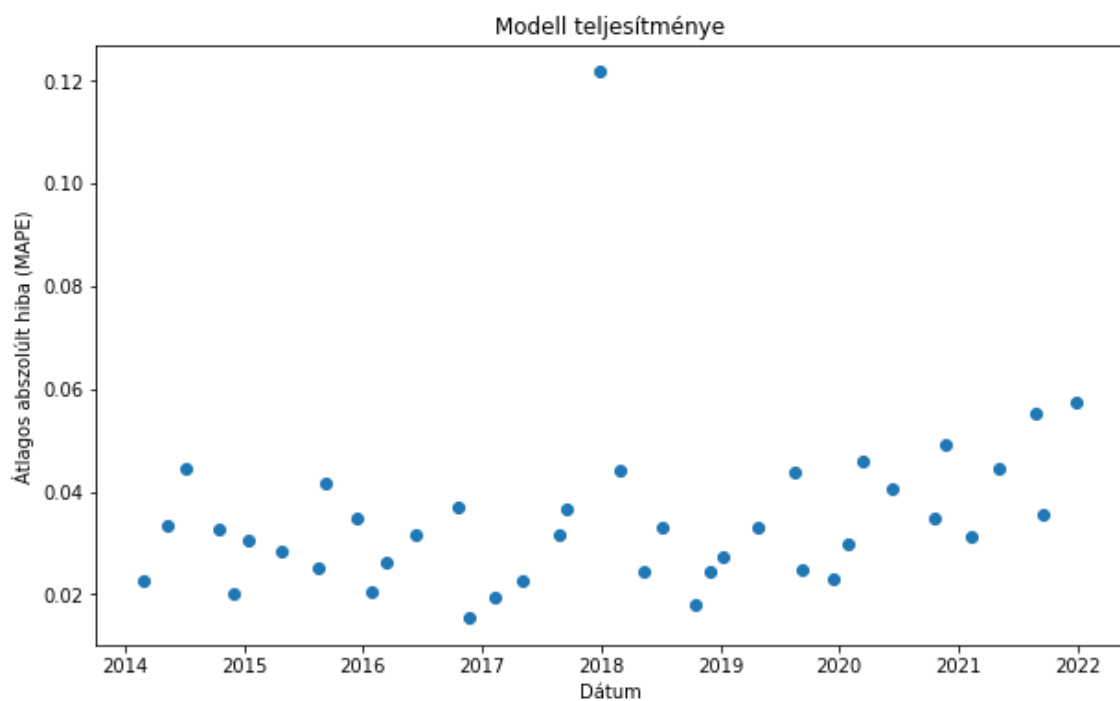
Az adatok előfeldolgozását is megváltoztattam. Az új megoldás értelmében az aggregált adatok hozzárendelésekor, ha egy munkanapot megelőző időszak munkaszüneti nap volt, akkor az adatponthoz az egy héttel korábbi adatok rendelődnek. Például, amennyiben az egyik év március 15. nap utáni 48 óra fogyasztását akarjuk prediktálni, akkor bemenetként nem a március 15. nap fogyasztási adatait használjuk fel bemenetként, hanem az egy héttel ezelőtti nap adatait.

Ezzel a megoldással előkészített adathalmaz esetében egy RandomForest regressziós modellt tanítottam fel, amely az előző modellekhez hasonló paraméterekkel rendelkezett. Ez a modell az alapmodellhez képest átlagosan 0,5%-kal jobb eredményt

szolgáltatót, de az első modell által nem jól prediktált adatpontot ez sem tudta jól előre jelezni.

	Átlag	Minimum	Maximum	Medián
<b>MSE</b>	61624,319473	11795,827072	461245,567843	42367,986315
<b>MAE</b>	178,49086	84,236750	573,651165	164,86916
<b>MAPE</b>	0,03492	0,015508	0,121627	0,032157

6.5. táblázat: A modell hibametriái munkarend felhasználása esetén



6.4. ábra: A modell átlagos abszolút százalékos hibája munkarend felhasználása esetén

## 6.6 Késleltetett fogyasztást alkalmazó modell

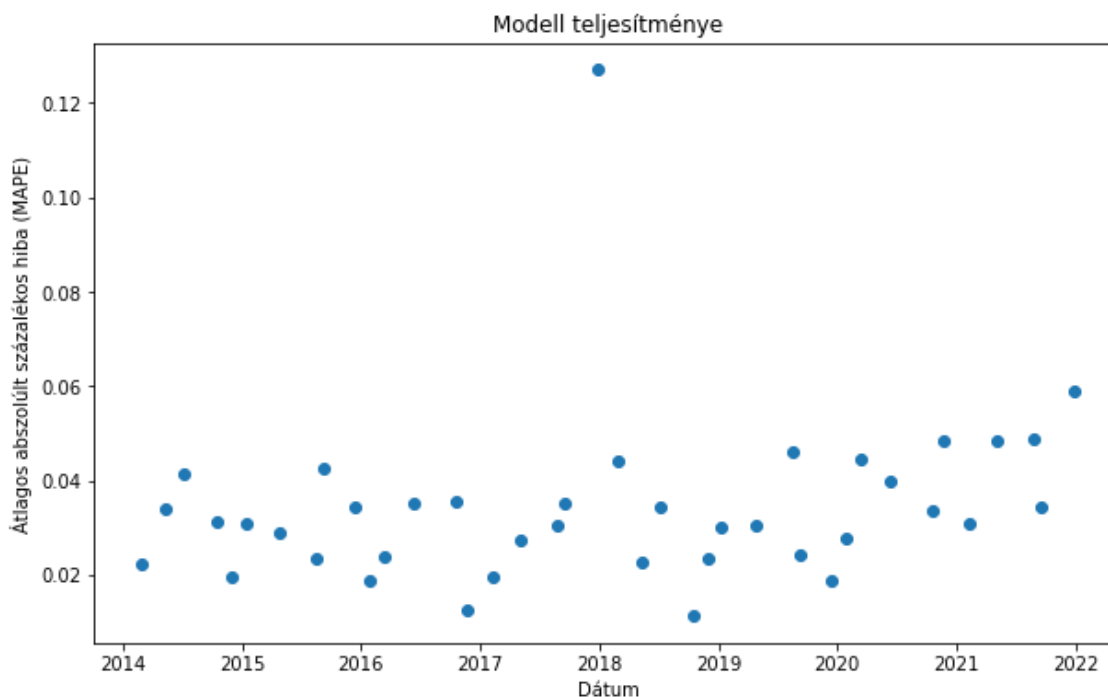
Az idősoros adatelemzési feladatok gépi tanulási elven végrehajtott módszereinek egyik jellegzetességé, hogy az aktuális célváltozó értéke helyett speciális mesterséges bemenő adatok segítségével tanítják fel a modellt. A statisztikai modellek a korábbi célváltozó értékek segítségével próbálják megjósolni a várható kimenetet.

A következő modell során megpróbáltam ötvözni ezt a két megközelítést és a bemeneti adathalmazba felvettem egy új adatoszlopot, ami az N órával ezelőtti fogyasztási értéket tartalmazza. Ezzel azt próbáltam elérni, hogy a korábbi időszak jellegzetességeit felismerve hatékonyabb modellt kapjak. Amennyiben N értéke osztható 24-gyel, akkor a fogyasztás egy előző nap hasonló időszakára esik, ezért viszonylagos hasonlóságot kell mutatnia a célváltozó értékével.

Egy RandomForest alapú regressziós modellt tanítottam fel a korábbiakhoz hasonló paraméterezéssel a fentiekben részletezett bemeneti adathalmaz felhasználásával. Azt tapasztaltam, hogy az előző modellhez képest ez a megközelítés sem hozott jelentős javulást, kb. 0,05%-kal emelte a teljesítményt. A korábbi modelleknél tapasztaltakhoz hasonlóan a szélsőséges karácsony körüli időszakot ez a modell sem tudta pontosan megbecsülni.

	<b>Átlag</b>	<b>Minimum</b>	<b>Maximum</b>	<b>Medián</b>
<b>MSE</b>	59962,541529	7180,419456	484977,049703	41677,876697
<b>MAE</b>	175,824839	60,635256	598,400792	163,049275
<b>MAPE</b>	0,034397	0,011386	0,126999	0,031085

6.6. táblázat: A modell hibametrikái késleltetett fogyasztás felhasználása esetén



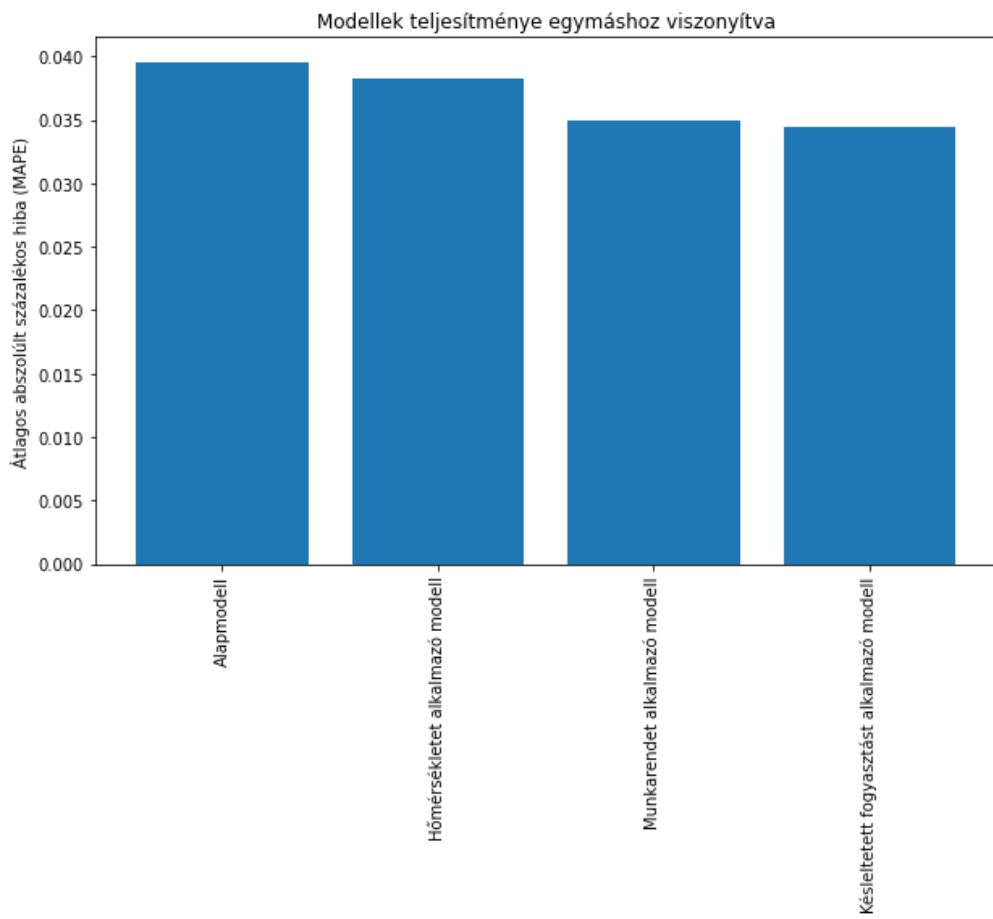
6.5. ábra: A modell átlagos abszolút százalékos hibája késleltetett fogyasztás felhasználása esetén

## 6.7 Modellek összegzése

Azt tapasztaltam, hogy az egyes modellek mindig egy kicsivel túlteljesítették az előző modellt, de az alapmodellhez képest túl nagy javulást nem sikerült elérni egyik esetben sem.

Minden modellre igaz volt, hogy a 2016.12.28. dátumot követő 48 óra fogyasztását nem tudta jól előrejelezni. Az is megfigyelhető volt, hogy míg az átlagos időszakok esetében az egyre fejlettebb modellek egyre pontosabb eredményt szolgáltatottak, addig ezt a különös időszakot egyre pontatlanabban jósolták meg.

Ez a nagy eltérés azzal magyarázható, hogy a karácsonyi időszak utáni munkanapokon általában nem üzemel teljes kapacitással a magyar ipar és gyakran csak januárban áll vissza a gazdaság a teljes kapacitására. Ezért ezen időszak fogyasztása nagyon eltér a szokványostól, a modellek nem képesek ezt pontosan megtanulni. A problémára megoldás lehet, hogy az ünnepnapokhoz közeli időszakok kezelésére speciális modelleket készítünk, amit csak ünnepnapokhoz közeli időszakok adatai alapján tanítunk fel.



**6.6. ábra: Modellek átlagos százalékos hibáinak összehasonlítása**

## 7 A modell interpretációja

A gépi tanulási módszerek alkalmazásakor napjainkban felmerülő igen fontos kérdés, a modell interpretációja. A mesterséges intelligenciával támogatott gépi tanulási folyamatok esetén fontos, hogy megérthető legyen, az algoritmus miért szolgáltatja a kimeneteket.

Az üzleti döntések végső meghozataláért mindig emberek a felelősek, jelenleg egy döntés következményeit nem lehet az algoritmusra hárítani. Ezért nagyon fontos, hogy a döntéshozók tisztában legyenek a gépi tanulási módszerek működésével, szükség esetén döntéseiket indokolni tudják a vitás kérdésekben. [14]

Az interpretáció mellett egy másik fő érv, hogy a modell által feltárt összefüggéseket felhasználva az adott szakterület humán szereplői is új tudásra tehessenek szert. A gépi tanulási algoritmusok gyakran fedeznek fel az adatok között az ember számára nem nyilvánvaló összefüggéseket is. [15]

A modellek implementációja esetén a legfontosabb változók kiválasztására egy gyakran használt eljárás, hogy először az összes attribútum felhasználásával feltanítanak egy modellt. Ezt interpretálva megállapítják, hogy mely változók a legrelevánsabbak a kimenet szempontjából és a végső modell tanításánál ezeket a bemeneti változókat használják fel. [14]

Ez az eljárás humán szereplők által készített összetett modellek egyszerűsítésére is felhasználható ilyenkor a gépi tanulás segítségével megpróbálnak egy modellt feltanítani, majd ezt interpretálják. Előfordulhat, hogy a gép egy sokkal egyszerűbb modellt képes alkotni, mint a kiindulási emberi modell. [15]

Általában minél hosszabb távra szóló döntéseket kell meghozni, annál fontosabb, hogy a modellek magyarázhatóak legyenek. Probléma viszont, hogy általában a legnagyobb pontosságú legkomplexebb modellek a legkevésbé interpretálhatóak, míg az egyszerűbb, de kisebb hatékonyságú modellek viszont jól magyarázhatóak. Amennyiben a magyarázhatóság egy fontos szempont, akkor meg kell találni az egyensúlyt a magyarázhatóság és a pontosság között. [14] [15]

## 7.1 A PDP grafikonok használata modellek interpretációjához

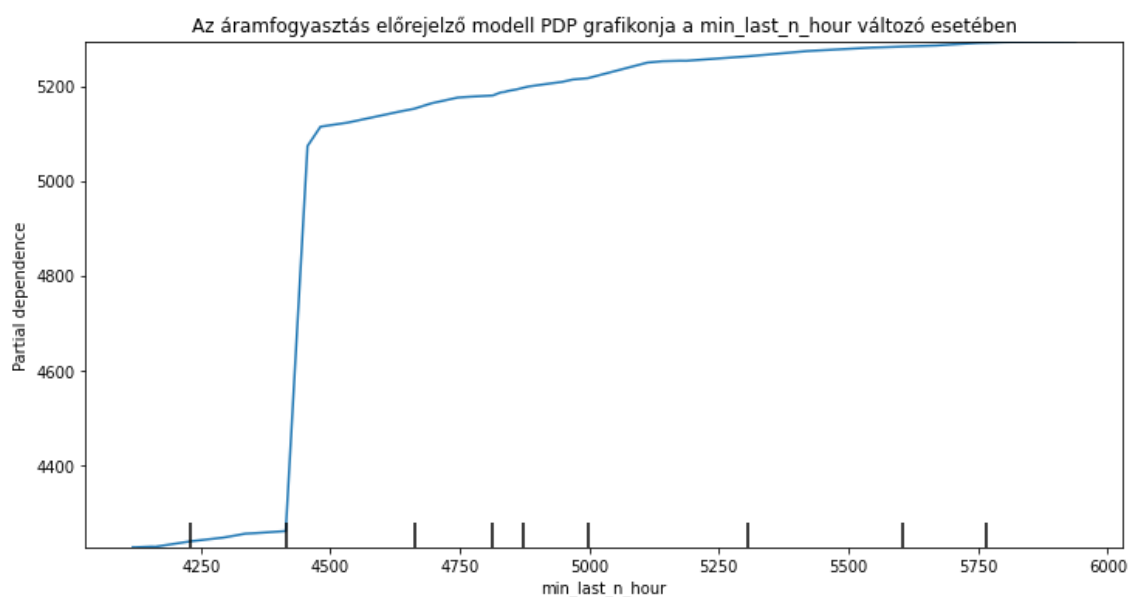
A modellek interpretációjához használt egyik legáltalánosabb módszer az úgynevezett Partial Dependence Plot-ok használata. A PDP plotok a globális modell interpretációs módszerek családjába tartoznak. A teljes adathalmazt felhasználva az egyes bemeneti változók által a kimeneti változóra gyakorolt hatását grafikusán ábrázolják.[14]

A PDP plot-ok egy egyszerűen érthető ábrán jelenítik meg a modell egyes változói és a kimenet közötti összefüggéseket. Ekkor a többi változóról egy átlagos értéket feltételezünk. A megközelítés azt feltételezi, hogy az egyes bemeneti változók egymástól függetlenek, ami általában nem igaz. Ezért fenntartással kell kezelni az ábráról leolvasott eredményeket. [13]

## 7.2 Áramfogyasztási modellek értelmezése

Az általam készített áramfogyasztási modellt is PDP grafikonok alapján értékeltem és próbáltam kapcsolatot keresni az egyes attribútumok és kimenet értékei között.

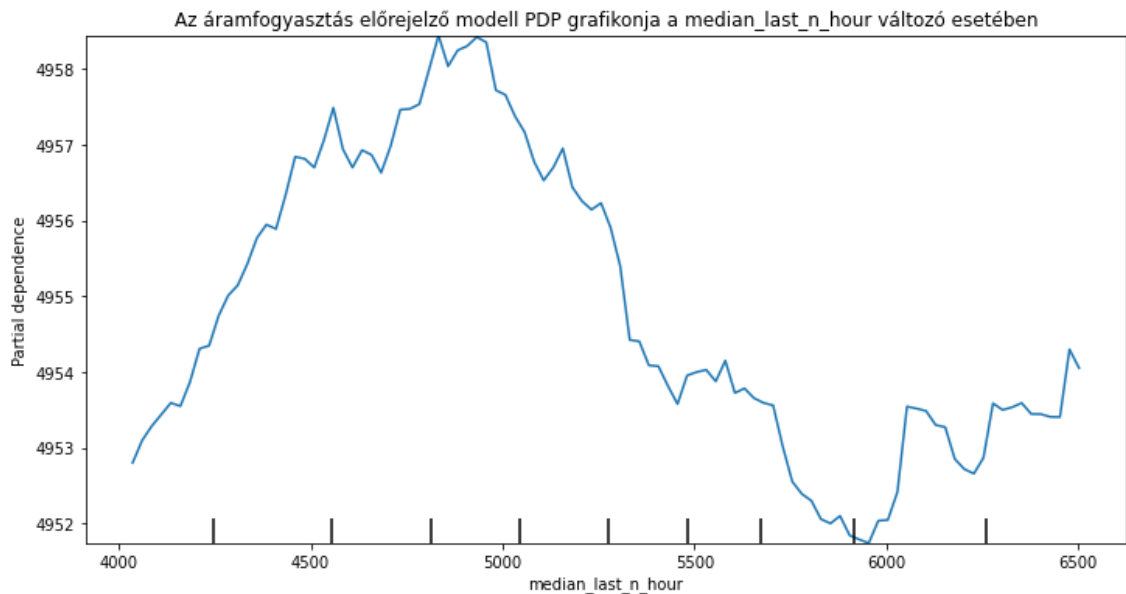
Elsőként az áramfogyasztás és az elmúlt N óra minimum fogyasztása közötti összefüggéseket vizsgáltam meg. Itt azt tapasztaltam, hogy a változó folyamatos növekedése esetén 4375 MWh alatt lassan növekszik a célváltozó értéke is. A 4375 MWh értéknél egy hirtelen ugrás látható, majd ezután a pont után szintén lassú növekedés észlelhető.



7.1. ábra: A min\_last\_n\_hour bementi változó PDP plotja

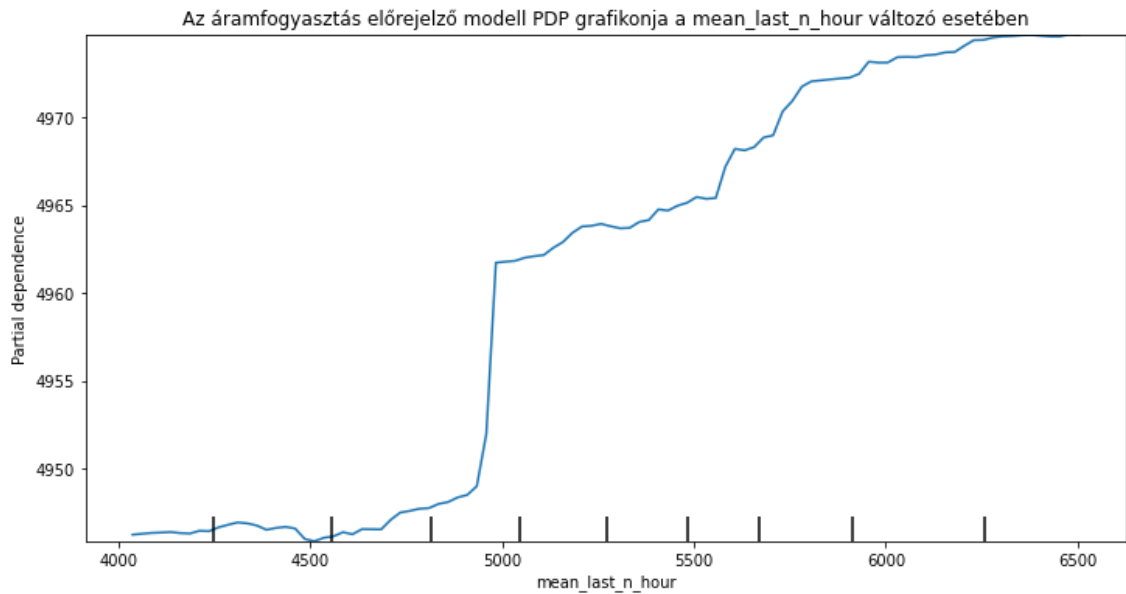


Vizsgáltam a fogyasztás és az elmúlt N óra fogyasztásának a medián értéke közötti összefüggést is. Itt a bemeneti változói értékeinek növekedése esetén többféle trend figyelhető meg a kimeneti változó esetében. Kb. 4750 MWh óra fogyasztásig az aktuális fogyasztás értéke is növekszik, de ezután a pont után csökkenésbe kezd a kimeneti változó értéke. 6000 MWh óra medián fogyasztás után kismértékben újra nő a célváltozó értéke.



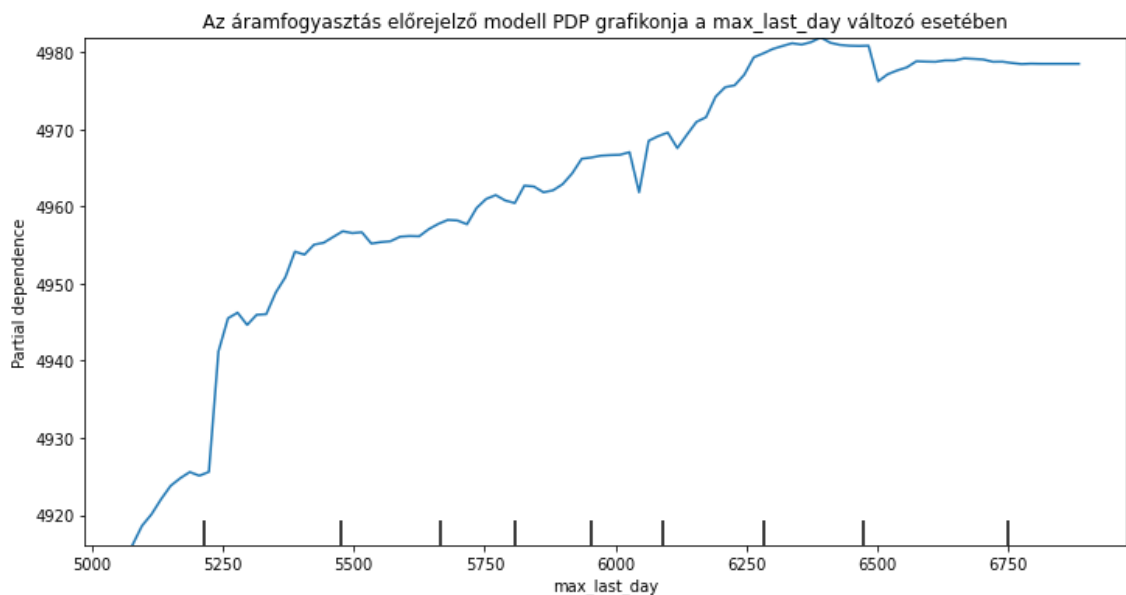
**7.2. ábra: A median\_last\_n\_hour bemeneti változó PDP plotja**

Megvizsgáltam az elmúlt N óra átlagos fogyasztása és a fogyasztás közötti összefüggéseket is. Azt tapasztaltam, hogy kb. 4500 MWh átlagos fogyasztásig az aktuális fogyasztás értéke közel konstans, majd egészen 5000 MWh átlagfogyasztásig ez lassan emelkedik. Ezután egy hirtelen ugrást láthatunk, amit egy meredekebb emelkedő szakasz követ, amely intenzitása lassan csillapodik, míg végül konstans értéket nem vesz fel.



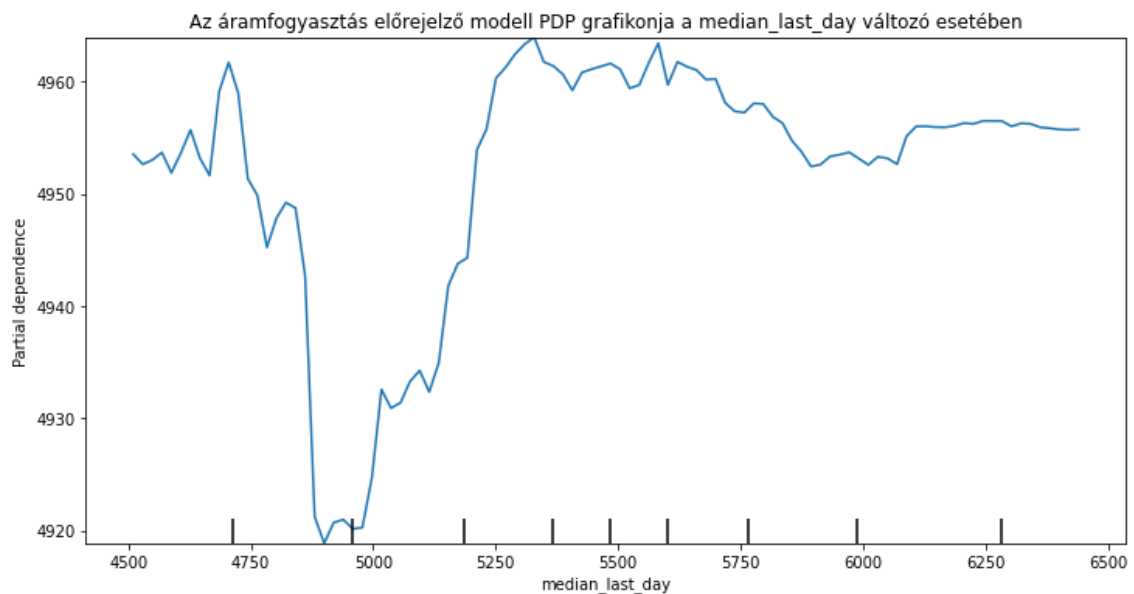
**7.3. ábra: A mean\_last\_n\_hour bemeneti változó PDP plotja**

Következőnek az előző 2 nap maximum fogyasztását vizsgáltam meg. Itt azt tapasztaltam, hogy a bemeneti változó folyamatos növekedése esetén kb. 5100 MWh szintig egyenletes változik a kimeneti változó értéke, majd ezen értéknél egy nagy ugrást tapasztalunk. Ezután viszonylag meredekebb módon követi le a célváltozó értéke a bemeneti változó értékváltozását, majd a nagyobb értékek esetén lassan csökken a növekedés intenzitása, végül pedig konstans értékre áll be.



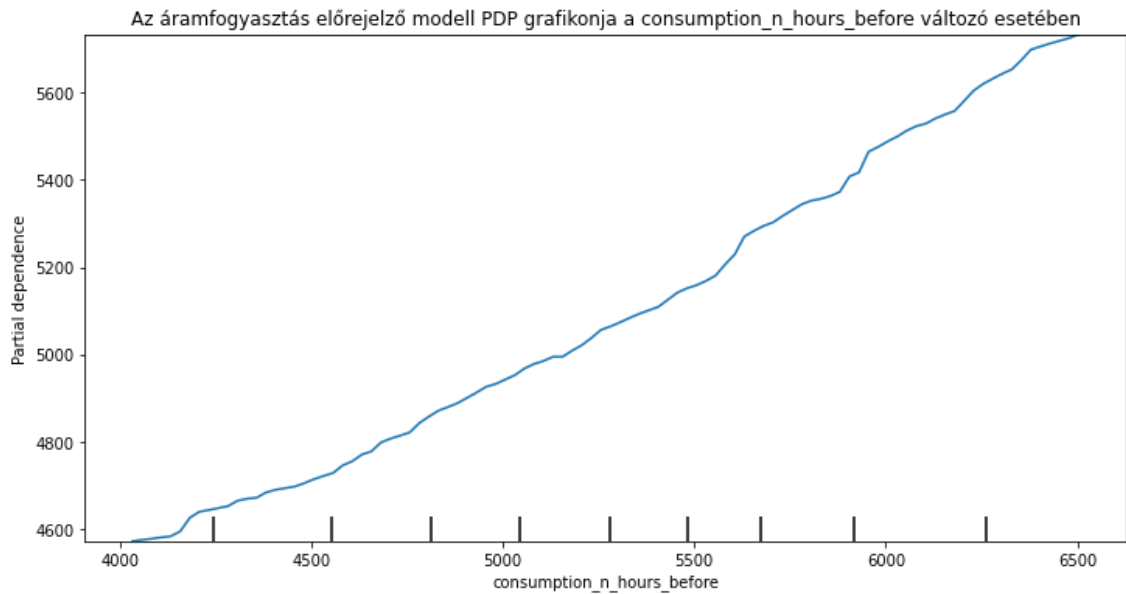
**7.4. ábra: A max\_last\_day bemeneti változó PDP plotja**

Ezt követően az elmúlt 2 nap medián fogyasztása és a fogyasztás összefüggését vizsgáltam. Ebben az esetben azt tapasztaltam, hogy az elmúlt nap medián fogyasztása és a fogyasztás közötti kapcsolat kifejezetten meredeken változik. Először egy csökkenő trend figyelhető meg kb. 5000 MWh értékig, ezt egy meredek emelkedési időszak követ egészen 5500 MWh óra értékig. Ezután egy kevésbé intenzív csökkenő szakasz következik, majd végül nagyjából konstans értéket vesz fel a célváltozó értéke.



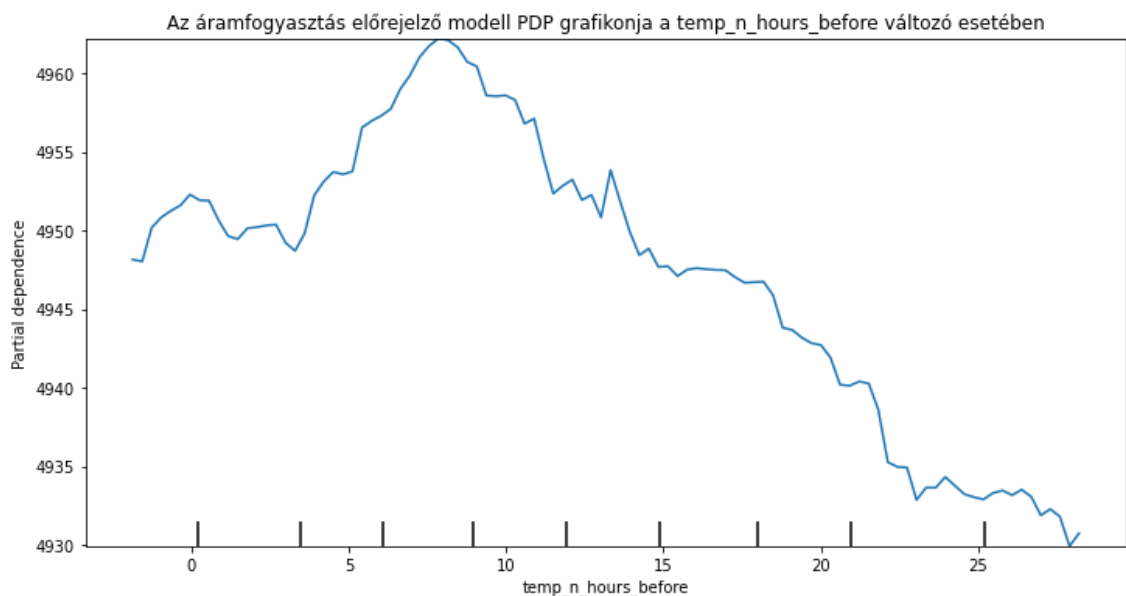
**7.5. ábra: A median\_last\_day bemeneti változó PDP plotja**

Ezt követően azt vizsgáltam, hogy milyen kapcsolat lehet a fogyasztás és a 48 órával ezelőtti fogyasztás között. Az ábrán lineáris kapcsolat figyelhető meg a két érték között. Ez az észrevétel reális, mivel az áramfogyasztás az idő függvényében egy viszonylag stabil periodicitást mutat.



**7.6. ábra: A consumption\_n\_hours\_before bemeneti változó PDP plotja**

Végül pedig a 48 órával ezelőtti hőmérséklet és a fogyasztás közötti kapcsolatot vizsgáltam meg. Itt azt tapasztaltam, hogy az alacsony környezeti hőmérsékleti értékek esetén a fogyasztás növekszik, míg a magasak esetén pedig csökken. Ez összhangban van az adatok feltáró elemzése során megfogalmazott észrevételeimmel, miszerint a téli hónapokban sokkal nagyobb az áramfogyasztás, min a tavaszi vagy nyári időszakokban. Meglepő az adatokból leolvasható azon eredmény, hogy a fogyasztás és a hőmérséklet közötti kapcsolat nem lineáris. Ez az eredmény valószínűleg azért olvasható ki a modelltől, mert a változók nem teljesen függetlenek egymástól.



**7.7. ábra: A temp\_n\_hours\_before bemeneti változó PDP plotja**

## 8 Összefoglalás és továbbfejlesztési lehetőségek

Munkám során a magyarországi villamos-energia rendszerterhelésének idősoros előre jelzésének feladatával foglalkoztam. Különbféle gépi tanulási modelleket építettem fel a MAVIR Zrt. nyilvános adatai alapján. Ezen kívül országos hőmérsékleti és munkarendi adatokat is felhasználtam.

Több különféle modellt megvizsgáltam a probléma megoldására. Az általam készített legjobb machine-learning modell kb. átlagosan 3,5% hibával volt képes prediktálni a jövőbeli áramfogyasztás értékét.

Az összes modellre igaz volt, hogy viszonylag jó hatékonysággal képes volt a normál munkanapok fogyasztását meghatározni. A hosszabb, többnapos ünnepek közelében található napok esetében, különösen az év végi karácsonyi időszakban viszont nagyon rossz pontosságú előrejelzést szolgáltatottak.

Munkám továbbfejlesztéseként a legfontosabb feladat az lenne, hogy a modelleket úgy kellene optimalizálni, hogy az említett szélsőséges időszakokra is viszonylag pontos előrejelzést adjanak.

Jelenleg úgy tapasztaltam, hogy minden  $N$  paraméterre másfajta modelltípus és hiperparaméterek nyújtják a legjobb teljesítményt. Ezért további fejlesztési út lehetne egy olyan modell készítése, ami viszonylag jó predekciót adna az  $N$  óra paraméter értékétől függetlenül, annak minden értékére.

Még fontos kitérni az algoritmusok futtatási teljesítményére. Azt tapasztaltam, hogy minél későbbi dátumra próbálunk prediktálni, annál tovább tart az algoritmusok futási ideje. Ez a modellépítésnél a tanító adatok mennyiségével magyarázható, mivel a későbbi dátumok esetében több tanító adatot használunk a modell feltanításához.

A teljesítmény meghatározásához méréseket végeztem egy átlagos képességű számítógépen. Itt azt tapasztaltam, hogy a futtatás ideje attól függ, hogy a bázisdátum és a 2010-es év között hány év telt el. Egy év átlagosan 112 másodperccel járult hozzá a futási időhöz.

Összességében elmondható, hogy a gépi tanulási eljárásokkal jól meg lehet oldani az előrejelzési feladatot, amely segítségével az átlagos fogyasztási időszakok fogyasztását jó pontossággal előre tudjuk jelezni.

# Irodalomjegyzék

- [1] Dr. Bókey Béla, Dr Rácz László: Villamosenergia-rendszerek stabilitása. Műszaki Könyvkiadó, 1988
- [2] Dr. Geszti P. Ottó: Villamosenergia-rendszerek II. kötet. Tankönyvkiadó, 1984.
- [3] Dr. Stróbl Alajos: A magyar VER teljesítmény szabályozása
- [4] MET Magyarország Zrt, Energiapiaci betekintő Magyarország energiatermelése és fogyasztása, <https://hugas.met.com/hu/energiapiaci-betekinto/magyarorszag-energiatermelese-fogyasztasa-eromuvek/8>, megtekintve: 2022. 10. 24.
- [5] A MAVIR Zrt. publikus rendszerterhelési adatai, <https://www.mavir.hu/web/mavir/rendszerterheles>, megtekintve 2022. 10. 24.
- [6] ML Approaches for Time Series, <https://towardsdatascience.com/ml-approaches-for-time-series-4d44722e48fe>, megtekintve 2022. 10. 24.
- [7] Dinesh C. S. Bisht; Mangey Ram: On Dimensionless Dissimilarity Measures for Time Series, <https://ieeexplore.ieee.org/document/9796087>, megtekintve 2022. 10. 24.
- [8] Dr. Bodon Ferenc, Adatbányászati algoritmusok, Műegyetem kiadó, 2010
- [9] Scikit-Learn Documentation: Decision Trees, <https://scikit-learn.org/stable/modules/tree.html>, megtekintve: 2022. 10. 24.
- [10] Scikit-Learn Documentation: Ensemble Methods, <https://scikit-learn.org/stable/modules/ensemble.html>, megtekintve: 2022. 10. 24.
- [11] Scikit-Learn Documentation: Model Evaluation, [https://scikit-learn.org/stable/modules/model\\_evaluation.html](https://scikit-learn.org/stable/modules/model_evaluation.html), megtekintve: 2022. 10. 24.
- [12] Scikit-Learn Documentation: Feature selection, [https://scikit-learn.org/stable/modules/feature\\_selection.html](https://scikit-learn.org/stable/modules/feature_selection.html), megtekintve: 2022. 10. 24.
- [13] Scikit-Learn Documentation: Partial Dependence, [https://scikit-learn.org/stable/modules/partial\\_dependence.html](https://scikit-learn.org/stable/modules/partial_dependence.html), megtekintve: 2022. 10. 24.
- [14] A Brief Overview of Methods to Explain AI (XAI), <https://towardsdatascience.com/a-brief-overview-of-methods-to-explain-ai-xai-fe0d2a7b05d6>, megtekintve: 2022. 10. 24.
- [15] Dmlab: XAI, avagy a magyarázható mesterséges intelligencia, <https://dmlab.hu/blog/xai-avagy-a-magyarezható-mesterseges-intelligencia/>, megtekintve: 2022. 10. 24.