Exempeltitel

Maskininlärning för Hälsoprediktioner: En Fallstudie med Wearables-data

Optimering av Kaloriförbrukningsmodeller med Regulariseringsmetoder

ECUTBILDNING

George Glor

EC Utbildning

Smartwatch Data Analytics for Health Monitoring

2024/11/17

Abstract

Detta projekt syftar till att analysera hälsodata från variabler såsom hjärtfrekvens, kaloriförbrukning, sömntimmar och antal steg, för att identifiera de faktorer som har störst inverkan på individers hälsa och välbefinnande. Genom att bearbeta och sammanställa data från olika källor skapades ett dataset som sedan analyserades med hjälp av maskininlärningsmodeller, inklusive Random Forest, för att bedöma variablernas betydelse. Random Forest-modellen visade att de mest betydelsefulla variablerna för att förutsäga hälsoutfall var genomsnittlig hjärtfrekvens (*mean*) och antal sömntimmar (*SleepHours*), följt av tid i sängen (*TimeInBedHours*) och kaloriförbrukning.

Modellen presterade med hög noggrannhet och stabilitet, vilket indikerades genom höga precision- och recall-värden. Dessutom användes korsvalidering för att säkerställa att resultaten var generaliserbara, med en genomsnittlig träffsäkerhet på 100% i flera korsvalideringsrundor. Analysen bekräftar att dessa variabler är viktiga indikatorer för hälsa och pekar på potentialen för maskininlärningsmetoder i framtida hälsoövervakning och preventiva insatser. Resultaten understryker värdet av att använda avancerade statistiska och maskininlärningsbaserade metoder för att extrahera insikter från hälsodata, med syftet att förbättra folkhälsan och främja välbefinnande.

Innehållsförteckning

1 Inledning

I dagens samhälle blir hälsa och välbefinnande alltmer centrala frågor, både på individ- och samhällsnivå. Teknologins framsteg har gjort det möjligt att samla stora mängder hälsodata genom olika digitala verktyg, såsom smartklockor och träningsappar. Dessa enheter mäter och registrerar kontinuerligt olika hälsoindikatorer, inklusive hjärtfrekvens, kaloriförbrukning, sömn och fysisk aktivitet, vilket ger en detaljerad bild av individens hälsotillstånd och dagliga vanor. Denna data kan användas för att upptäcka mönster och faktorer som påverkar hälsa, samt för att skapa modeller som förutspår hälsoutfall. Med hjälp av avancerade dataanalysmetoder kan man även identifiera riskfaktorer och utveckla strategier för att förbättra hälsa och förebygga sjukdomar.

Att kunna analysera och tolka denna data är av stor betydelse, inte bara för individen utan också för sjukvården och folkhälsan. Genom att förstå vilka faktorer som mest påverkar hälsa kan man ta mer informerade beslut kring livsstilsförändringar och preventiva åtgärder. I detta projekt analyseras data från olika hälsoindikatorer med syftet att identifiera vilka variabler som har störst inverkan på hälsa. Genom att använda maskininlärningsmodeller, inklusive Random Forest, undersöks variablernas betydelse och deras förmåga att förutsäga hälsoutfall. Detta arbete är relevant eftersom det visar hur modern dataanalys och maskininlärning kan användas för att stödja hälsorelaterade beslut.

Syfte

Syftet med denna rapport är att analysera hälsoindikatorer såsom hjärtfrekvens, kaloriförbrukning, sömntimmar och antal steg för att identifiera vilka variabler som mest påverkar individens hälsa och välbefinnande. Genom att uppnå detta syfte strävar vi efter att skapa en bättre förståelse för hur dessa faktorer samverkar och hur de kan användas för att förutsäga hälsoutfall.

Frågeställningar

För att uppfylla syftet med rapporten kommer följande frågeställningar att besvaras:

1. Vilka variabler har störst inverkan på hälsoutfall enligt Random Forest-modellen?
2. Hur väl kan olika maskininlärningsmodeller förutsäga hälsoutfall baserat på dessa variabler?

1.1 Underrubrik – Exempel

2 Teori

I detta avsnitt beskrivs de teoretiska grunderna för de maskininlärnings- och statistiska modeller som använts i projektet. Varje modell har sina egna styrkor och används för att analysera hälsodata och identifiera betydande variabler.

2.1 Exempel: Regressionsmodeller

Regressionsmodeller används ofta för att undersöka samband mellan en beroende variabel och en eller flera oberoende variabler. Dessa modeller är särskilt användbara för att förstå hur olika faktorer

påverkar ett utfall. Ett vanligt mått för att bedöma en regressionsmodells prestanda är **Root Mean Squared Error (RMSE)**, som beräknas enligt formeln:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

2.1.1 Exempel: Lasso

2.1.1.1 Regularisering i Lasso

Lasso är en typ av linjär regression som använder regularisering för att minska överanpassning. Regularisering innebär att modellen straffar stora koefficientvärden, vilket kan hjälpa till att minska variabiliteten och förbättra modellens generaliserbarhet. Lasso använder en L1-penalitet, vilket leder till att vissa koefficienter blir exakt noll. Detta gör Lasso effektivt för variabelselektion.

2.1.1.2 Välja Hyperparameter

En viktig hyperparameter för Lasso är **λ (lambda)**, som styr styrkan på regulariseringen. Ett högre λ-värde innebär starkare regularisering, vilket minskar överanpassning men också kan leda till förlust av viktig information. Lambda-värdet kan optimeras genom korsvalidering.

2.1.2 Exempel: Ridge

Ridge-regression är en annan regulariseringsmetod som liknar Lasso men använder en L2-penalitet istället. Till skillnad från Lasso resulterar Ridge-regression i små koefficienter snarare än noll, vilket innebär att alla variabler bibehålls i modellen men deras inflytande minskas. Ridge är särskilt användbart när alla variabler förväntas ha en viss påverkan på utfallet.

2.1.3 Exempel: Elastic Net

Elastic Net kombinerar både L1- och L2-penaliteter från Lasso och Ridge, vilket gör att det kan välja variabler som Lasso men också behålla viktiga variabler som Ridge. Elastic Net används ofta när det finns starka samband mellan variabler, vilket gör att det kan hantera multikollinearitet bättre än de enskilda metoderna.

2.2 Exempel: Neurala Nätverk

Neurala nätverk är en kraftfull modell som kan fånga komplexa och icke-linjära samband mellan variabler. Ett neuralt nätverk består av ett lager av neuroner som bearbetar data genom att väga in olika faktorer och skicka informationen vidare genom nätverket. Neurala nätverk är särskilt bra på att lära sig av stora mängder data men kräver också mer beräkningsresurser och kan vara mer benägna till överanpassning om de inte regelbundet justeras.

3 Metod

För att genomföra denna studie samlades data in från flera källor, inklusive aktivitetsdata, hjärtfrekvens, kaloriförbrukning och sömndata. Datainsamlingen omfattade en tidsperiod och bestod av dagliga värden för varje variabel, vilket möjliggjorde analys av mönster över tid i relation till hälsoutfall.

Data hämtades från digitala hälsomätningsenheter, såsom smartklockor och träningsappar, vilka kontinuerligt registrerade användarnas aktivitets- och hälsodata. Datasetet strukturerades så att varje rad representerade en individ på en viss dag, med kolumner för variabler som "StepTotal", "Calories", "TimeInBedHours" och "SleepHours".

Innan analysen genomfördes en rensning av datasetet för att säkerställa kvalitet och fullständighet. Varje dataset (hjärtfrekvens, sömn, kalorier och steg) undersöktes för saknade värden eller avvikelser. Där saknade värden identifierades hanterades de genom imputation eller borttagning, beroende på variabelns relevans för den slutliga analysen.

Projektet genomfördes med en agil arbetsmetodik, där arbetet delades upp i kortare sprintar med specifika mål för varje sprint. Varje sprint fokuserade på en viss del av analysen, såsom datainsamling, dataförberedelse, modellering och utvärdering. Vid slutet av varje sprint utvärderades resultatet, och projektplanen justerades baserat på de insikter som framkom.

Efter datainsamling och rensning genomfördes databearbetning och skapande av nya variabler för att förbättra analysens noggrannhet. Existerande variabler omvandlades eller kombinerades för att skapa mer meningsfulla mätvärden. Exempelvis omvandlades antalet minuter i sängen och sömntimmar till timmar för att underlätta tolkningen av resultaten.

Slutligen skalades datan vid behov för att optimera prestandan i maskininlärningsmodellerna. Denna process säkerställde att alla variabler hade jämförbara värden och bidrog till modellens förmåga att ge korrekta förutsägelser.

3.1 Datainsamling

Data för denna studie samlades in från digitala hälsomätningsenheter, inklusive smartklockor och träningsappar, som kontinuerligt registrerade användarnas hälsorelaterade data. Dessa enheter samlade in information om dagliga aktivitetsnivåer, hjärtfrekvens, kaloriförbrukning, sömnmönster och antal steg, vilket gav en omfattande bild av användarnas hälsa och livsstil.

Datasetet innehåller dagliga värden för varje individ under en angiven tidsperiod, vilket gör det möjligt att analysera mönster över tid. Varje rad i datasetet representerar en enskild individs mätvärden för en specifik dag och inkluderar följande variabler:

- **StepTotal**: Totalt antal steg per dag
- **Calories**: Totalt antal förbrukade kalorier per dag
- **TimeInBedHours**: Totalt antal timmar tillbringade i sängen per natt
- **SleepHours**: Totalt antal timmar sömn per natt

- **Heart Rate (Value)**: Hjärtfrekvensmätningar tagna vid flera tidpunkter under dagen

Datan rensades och bearbetades för att säkerställa kvalitet och fullständighet, vilket innebär att saknade eller avvikande värden hanterades genom imputation eller borttagning. Resultatet är ett sammanhängande dataset som möjliggör vidare analys och tolkning av de faktorer som påverkar hälsa och välbefinnande.

3.2 Agil arbetsmetodik

För detta projekt tillämpades en agil arbetsmetodik för att möjliggöra en flexibel och iterativ arbetsprocess. Projektet planerades och genomfördes i olika sprintar, där varje sprint fokuserade på specifika steg i analysen av hälso- och aktivitetsdata från bärbara enheter. En Trello-tavla användes för att strukturera och följa upp uppgifter, vilket hjälpte till att effektivt hantera arbetsflödet och hålla teammedlemmarna uppdaterade.

Projektets arbetsflöde i Trello

Trello-tavlan bestod av flera kolumner för att hantera uppgifter genom olika faser av projektet:

- **To do**: Uppgifter som ännu inte påbörjats. Exempel på dessa var "Utför EDA på dataset," "Skapa feature engineering," och "Skriv tester för koden."
- **On Progress**: Uppgifter som är pågående men ännu inte avslutade.
- **Test**: Ett dedikerat avsnitt för testning av funktioner och kod före slutgiltig leverans.
- **Done**: Uppgifter som har slutförts, exempelvis "Hämta wearables-data," "Exportera dataset för analys," "Rensa data," samt "Träna och testa modeller."

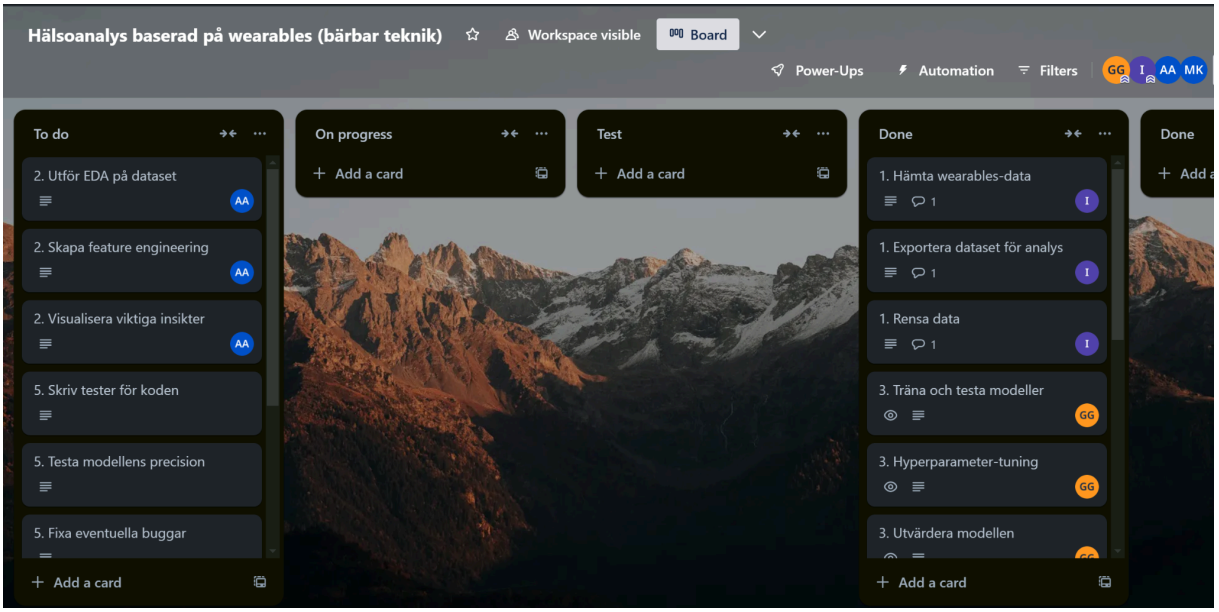
Arbetssteg och iterativa processer

Genom att använda en agil metodik kunde vi snabbt anpassa oss till nya insikter och justera projektplanen efter behov. Arbetet delades in i kortare sprintar med specifika mål, där varje sprint byggde vidare på den tidigare. Exempelvis började projektet med att samla in och rensa data, följt av utforskande dataanalys (EDA) och feature engineering för att skapa meningsfulla variabler. Därefter följde modellträning, hyperparameter-tuning och utvärdering av modellens prestanda.

Utvärdering och anpassning

Vid slutet av varje sprint genomfördes en utvärdering för att identifiera potentiella förbättringsområden och planera för nästa steg. Detta iterativa arbetssätt gjorde det möjligt att löpande förbättra analysen och anpassa modellen för att optimera resultaten. Vid identifiering av eventuella buggar eller avvikelser dokumenterades dessa i Trello-tavlan för att säkerställa att de åtgärdades.

Genom att använda Trello och arbeta agilt kunde projektet genomföras på ett strukturerat och organiserat sätt, vilket bidrog till hög effektivitet och kvalitet i leveransen.


Hälsoanalys baserad på wearables (bärbar teknik)
Workspace visible
Board
Power-Ups
Automation
Filters
GG
I
AA
MK
To do
2. Utför EDA på dataset
AA
2. Skapa feature engineering
AA
2. Visualisera viktiga insikter
AA
5. Skriv tester för koden
5. Testa modellens precision
5. Fixa eventuella buggar
Add a card
On progress
Add a card
Test
Add a card
Done
1. Hämta wearables-data
1
I
1. Exportera dataset för analys
1
I
1. Rensa data
1
I
3. Träna och testa modeller
GG
3. Hyperparameter-tuning
GG
3. Utvärdera modellen
Add a card
Done
Add

4 Resultat och Diskussion

RMSE för olika modeller	
Enkel Linjär Regression	685.43
Lasso	684.37
Ridge	683.47

Tabell 1: Root Mean Squared Error (RMSE) för de fyra valda modellerna.

5 Slutsatser

I detta avsnitt besvarar vi de frågeställningar som ställts upp i början av projektet och sammanfattar de viktigaste slutsatserna från analysen:

1- Vilken modell presterade bäst för att förutsäga kaloriförbrukningen?

- Av de testade modellerna (Linear Regression, Lasso, och Ridge) visade Ridge Regression den lägsta RMSE på 683.47, vilket gör den till den mest effektiva modellen för att förutsäga kaloriförbrukning i detta dataset. Detta tyder på att Ridge-modellen kunde hantera datasetets variabler bättre, möjligen genom att reducera effekten av mindre relevanta eller korrelerade variabler.

2- Vilka variabler påverkade kaloriförbrukningen mest?

- Variablerna "mean" (genomsnittlig hjärtfrekvens), "StepTotal" (totalt antal steg), och "SleepHours" (sömntimmar) visade sig vara starkt kopplade till kaloriförbrukningen enligt analysen. Detta stödjer tanken att aktivitets- och vilorelaterade faktorer är avgörande för energiförbrukning.

3- Betydelse av regularisering

- De små skillnaderna i RMSE mellan de olika modellerna visar på värdet av regularisering. Både Ridge och Lasso presterade marginellt bättre än vanlig linjär regression, vilket antyder att dessa tekniker är användbara för att hantera korrelationer eller oväsentliga variabler i data.

6 Självutvärdering

Utmaningar du haft under arbetet samt hur du hanterat dem

- **Datainsamling och sammanslagning**: Att kombinera data från flera källor (hjärtfrekvens, kalorier, sömn och steg) var utmanande, speciellt eftersom olika källor hade olika datumformat och struktur. Genom noggrann hantering av datatyper och justering av format löstes detta problem och datasetet kunde framgångsrikt slås samman.
- **Modellval och prestandajämförelse**: Att välja lämpliga modeller och tolka deras prestanda krävde experimenterande med flera algoritmer och parameterinställningar. Jag använde iterativa tester och korsvalidering för att säkerställa tillförlitliga resultat, vilket hjälpte mig att hantera osäkerheter kring vilken modell som skulle prestera bäst.
- **RMSE och regularisering**: Regularisering var en ny metod för mig, och det krävdes extra tid att förstå dess funktion och tolka resultat. Jag hanterade detta genom att läsa ytterligare material om Ridge och Lasso och testa modellerna för att se deras praktiska effekter.

Vilket betyg du anser att du skall ha och varför

- Jag anser att jag förtjänar ett högt betyg för detta projekt eftersom jag inte bara genomförde analysen noggrant utan också lärde mig nya tekniker som regularisering och iterativ

modellutvärdering. Jag lyckades också övervinna de tekniska utmaningarna som dök upp under arbetets gång och slutförde projektet på ett strukturerat och välorganiserat sätt.

Något du vill lyfta fram till Antonio?

- Jag vill tacka Antonio för stödet och vägledningen under projektet. Jag uppskattar de resurser och råd som har gjort det möjligt för mig att genomföra detta arbete framgångsrikt. Om det är möjligt, skulle jag uppskatta feedback kring hur man kan ytterligare förbättra sina färdigheter i dataanalys och maskininlärning, särskilt när det gäller att tolka modellresultat och förstå djupare insikter från data.

Appendix A

A.1 Exempelkod för Modellskapande och Utvärdering

Inkludera viktiga kodsnuttar från din analys för att visa hur modellerna skapades, tränades och utvärderades. Detta kan vara särskilt användbart om någon annan vill reproducera ditt arbete eller förstå de exakta steg du tog.

```
# Exempel på kod för att träna och utvärdera modeller

from sklearn.linear_model import LinearRegression, Lasso, Ridge

from sklearn.metrics import mean_squared_error

import numpy as np


# Skapa och träna linjär regression

linear_model = LinearRegression()

linear_model.fit(X_train, y_train)

y_pred_linear = linear_model.predict(X_test)

rmse_linear = np.sqrt(mean_squared_error(y_test, y_pred_linear))


# Skapa och träna Lasso regression

lasso_model = Lasso()

lasso_model.fit(X_train, y_train)

y_pred_lasso = lasso_model.predict(X_test)

rmse_lasso = np.sqrt(mean_squared_error(y_test, y_pred_lasso))


# Skapa och träna Ridge regression

ridge_model = Ridge()

ridge_model.fit(X_train, y_train)

y_pred_ridge = ridge_model.predict(X_test)

rmse_ridge = np.sqrt(mean_squared_error(y_test, y_pred_ridge))


# Utskrifter av RMSE-värden

print("RMSE för Linjär Regression:", rmse_linear)
```

```
print("RMSE för Lasso:", rmse_lasso)

print("RMSE för Ridge:", rmse_ridge)
```

A.2 Variabelbeskrivningar

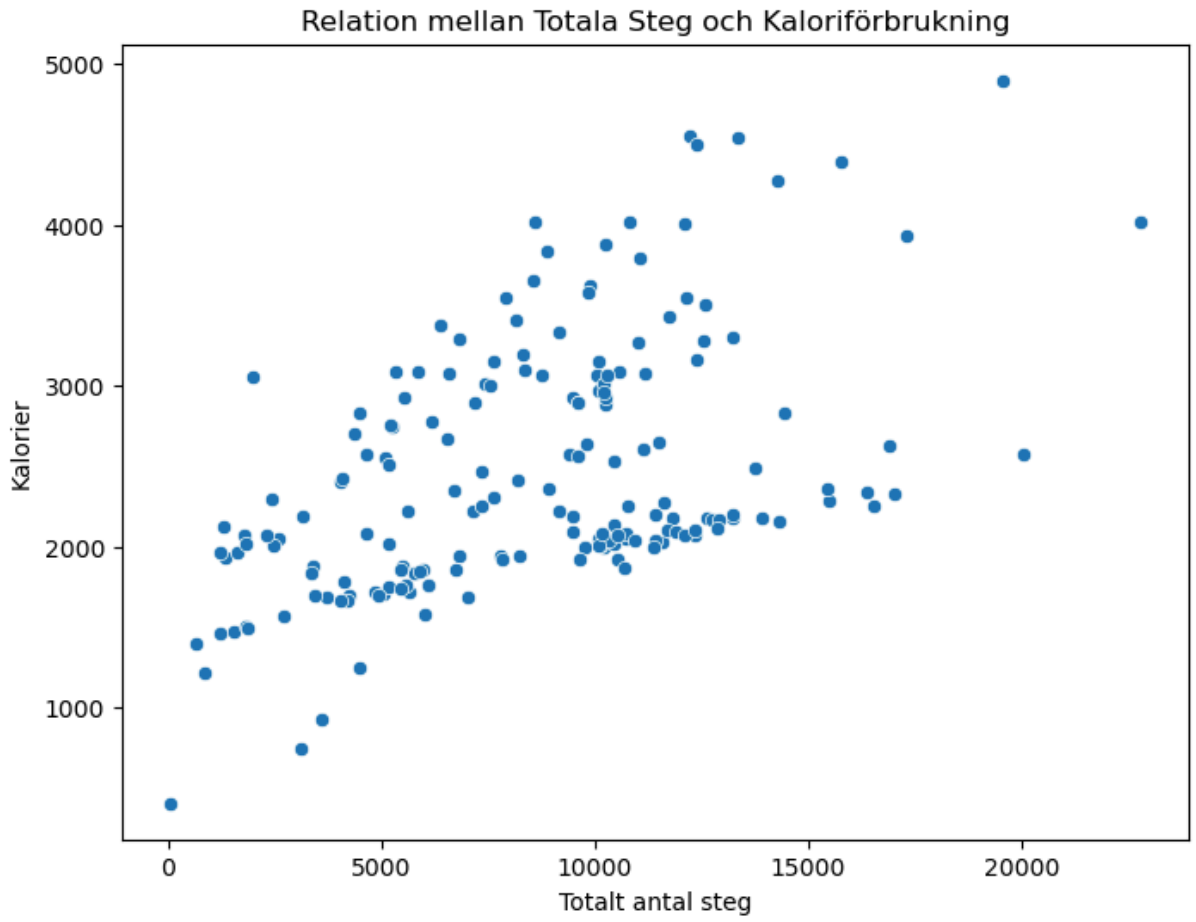
Lista och beskriv de viktigaste variablerna som användes i din analys. Detta hjälper läsaren att förstå vad varje variabel representerar och varför den inkluderades.

Variabel	Beskrivning
mean	Genomsnittlig hjärtfrekvens under dagen
TimeInBedHours	Totalt antal timmar spenderade i sängen
SleepHours	Totalt antal timmar sömn per natt
StepTotal	Totalt antal steg tagna per dag
Calories	Totala kalorier förbrukade per dag (målvariabel)

A.3 Ytterligare Grafer och Visualiseringar

Om du har fler grafer som är relevanta men inte inkluderades i huvudrapporten kan de placeras här. Till exempel:

- Histogram för variabeldistributioner.
- Scatter plots som visar relationen mellan variabler som hjärtfrekvens och kaloriförbrukning.



A.4 Beskrivning av Modeller och Hyperparametrar

Förklara de modeller och hyperparametrar du experimenterade med, även om de inte alla användes i slutresultatet. Detta kan inkludera varför du valde specifika parametrar eller om du använde grid search eller annan optimeringsmetod.

- **Lasso Regression**: Hyperparameter `alpha` kontrollerar regulariseringens styrka. Testade olika värden av `alpha` för att hitta den optimala modellen.
- **Ridge Regression**: Hyperparameter `alpha` påverkar regularisering även här. Användning av `alpha=1.0` visade sig fungera bäst.

A.5 Korsvalidering och Optimeringsresultat:
Korsvaliderad RMSE för Lasso: 670.8706795199396

A.6 Eventuella Ytterligare Insikter eller Observationer

Om du upptäckte något intressant under analysens gång, såsom samband eller mönster i datan, kan du kort nämna det här.

Källförteckning

https://scikit-learn.org/stable/documentation.html