Prognostisering av Framtida Utbildningskostnader

Girlie Razon

EC Utbildning

Projekt i Data Science

202411

Abstract

Detta arbete undersöker framtida utbildningskostnader för grundskola och gymnasieskola i Sverige mellan åren 2025 och 2035, med hänsyn till faktorer som befolkningsstruktur och demografiska trender. Genom att använda en ensemblemodell, där flera regressorer kombineras (bl.a. Linear Regression, Random Forest Regressor, Gradient Boosting Regressor och Voting Regressor), tillsammans med avancerade prognosmodel, LSTM, har vi identifierat övergripande trender och mönster i kostnadsutvecklingen. Ensemblemodellen har visat sig vara särskilt effektiv och tillförlitlig i att fånga upp förändringar i utbildningskostnader, vilket ger en robust grund för långsiktiga prognoser. Resultaten tyder på att kostnaderna per elev förväntas öka, drivet av regionala skillnader och förändringar i antalet elever. Denna studie bidrar med insikter som kan stödja beslutsfattande inom utbildningsfinansiering och resursfördelning för att möta framtidens behov.

Förkortningar och Begrepp

LSTM : Long Short-term Memory

Table of Contents

1 Inledning

Utbildning är en central del av samhällsutvecklingen och en viktig investering i framtida generationer. I Sverige är utbildning avgörande, inte bara för att utveckla individuella talanger och kompetenser utan också för att skapa en välutbildad arbetskraft som driver ekonomisk och social utveckling. De senaste årens förändringar i utbildningssystemet har skapat en ökad medvetenhet kring vikten av att förstå och förvalta utbildningskostnader, som varierar stort mellan regioner och skolnivåer. För att säkerställa en effektiv resursfördelning krävs därför tillförlitliga prognoser av framtida utbildningskostnader.

Dessa kostnader påverkar både kommuner och staten, vilket gör det nödvändigt att förutse kostnadsutvecklingen, särskilt med tanke på förändrade demografiska mönster som påverkar antalet elever och därmed behovet av resurser. Genom att prognostisera utbildningskostnader på grundskole- och gymnasienivå för perioden 2025–2035, kan regioner bättre planera och säkerställa att alla elever har tillgång till en kvalitativ utbildning.

I denna rapport används statistiska och ensemblebaserade modeller som Linear Regression, Random Forest Regressor och Gradient Boosting Regressor, som utvärderas med prestandamått som RMSE, MAE och R^2 för att skapa pålitliga prognoser. Målet är att besvara följande frågeställningar:

1. Vilka faktorer påverkar utbildningskostnaderna i olika regioner i Sverige?
2. Hur kan förändringar i befolkningsstrukturen påverka prognosen för utbildningskostnader?

Dessa frågor bidrar till en djupare förståelse för de faktorer som driver kostnader inom utbildningssektorn och hjälper beslutsfattare att optimera resursfördelning och anpassa sig till framtida behov.

(Observera att detaljer kring modellresultat och prestanda kommer att redovisas under avsnittet Resultat och Diskussion.)

2 Teori

Detta avsnitt introducerar den teoretiska bakgrunden som är relevant för att förstå sammanhanget av detta projekt.

2.1 Utvärderingsmått

Vid utvärdering av regressionsmodeller använder vi flera mått för att bedöma modellens prestanda och prognosprecision:

i. **RMSE** (Root Mean Squared Error): Ett mått på genomsnittligt fel där stora fel kvadreras vilket gör det känsligt för outliers.

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

ii. **MAE** (Mean Absolute Error): Genomsnittet av absolutvärdet av felen, vilket är robust mot outliers.

$$\boldsymbol{MAE} = \frac{1}{n}\sum_{i=1}^{n}|yi - \hat{y}i|$$

iii. **R^2** (R-Squared): Ett mått på hur mycket av variansen i målfunktionen som kan förklaras av modellen, värden nära 1 indikerar god förklaringsgrad.

$$\boldsymbol{R^2 = 1 = \frac{\sum_{i=1}^{n}(y_i - \hat{y}i)^2}{\sum_{i=1}^{n}(yi = \bar{y})^2}}$$

2.2 Regressionsmodeller för Prognos av Kostnader per Elev

Modellerna som används för att förutsäga utbildningskostnader över tid är huvudsakligen ensemble-baserade och linjära modeller.

2.2.1 Linear Regression

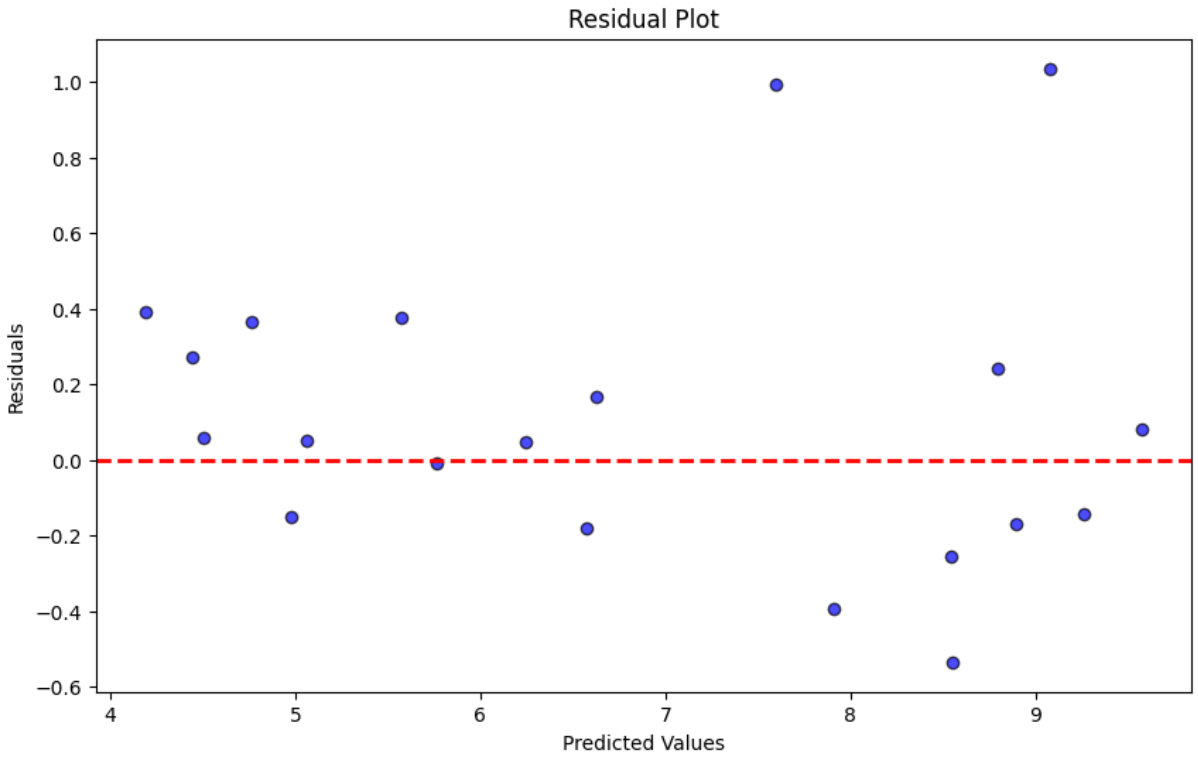
En enkel linjär modell som försöker hitta den linje som bäst passar datan genom minsta kvadratmetoden.

- Beräkning av residualer: **residuals = y_test – y_pred** beräknar skillnaderna mellan de faktiska och förutsagda värdena.
- Spridningsdiagram: Plotta **y_pred** på x-axeln och **residualerna** på y-axeln.

- Referenslinje: En horisontell linje vid **y=0** läggs till för att visuellt kontrollera om residualerna är centrerade kring noll.

I en välpassande linjär modell bör residualerna vara slumpmässigt spridda runt den horisontella linjen vid **y=0**. Dessutom bör inga uppenbara mönster (som en kurva) vara synliga i residualerna, mönster kan tyda på icke-linjärhet eller modellfel.

Figur 1: Spridningsdiagram



2.2.2 Random Forest Regressor

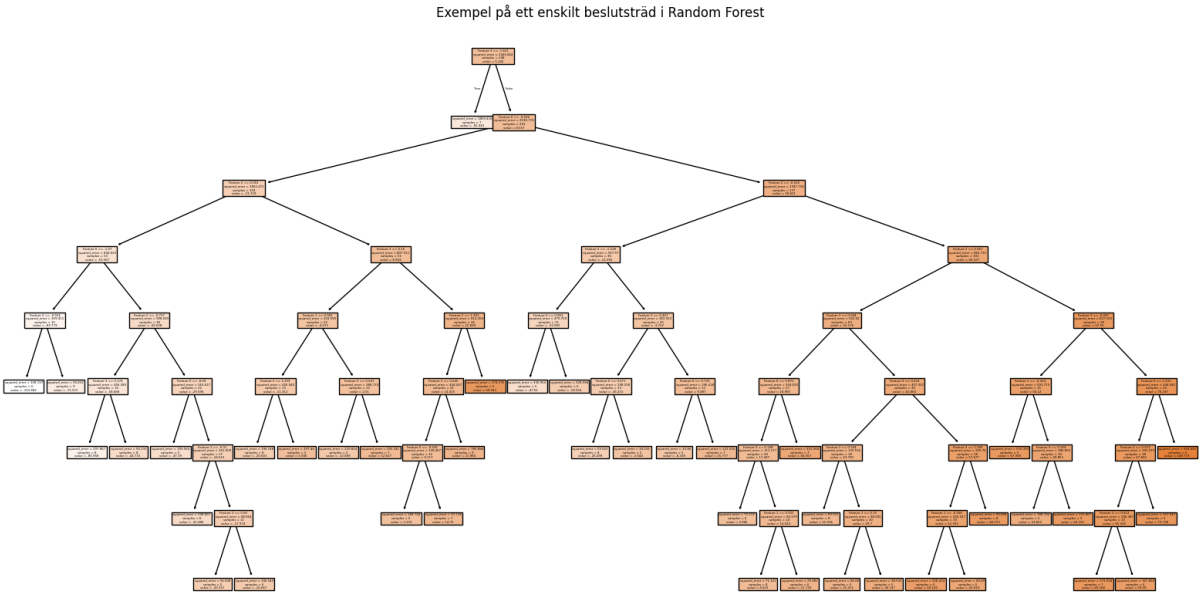
En ensemblemodell som bygger många beslutsträd och kombinerar dem för att förbättra prediktionen och minska överanpassning.

2.2.2.1 *Hyperparameter och Regularisering*

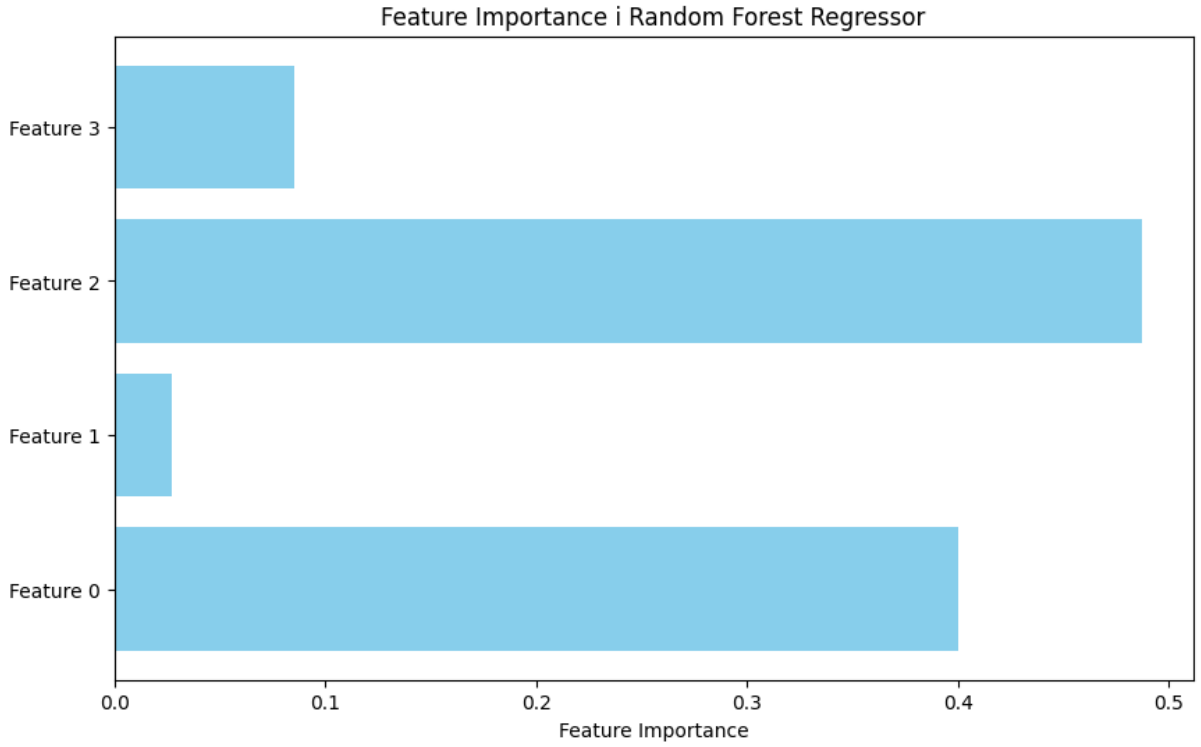
Viktiga hyperparametrar tuning som påverkar modellens prestanda inkluderar:

- **n_estimators**: Antal träd i skogen.
- **min_samples_leaf**: Minsta antal prover för att dela en nod.
- **max_features**: Andel av funktionerna som används för att bygga varje träd.

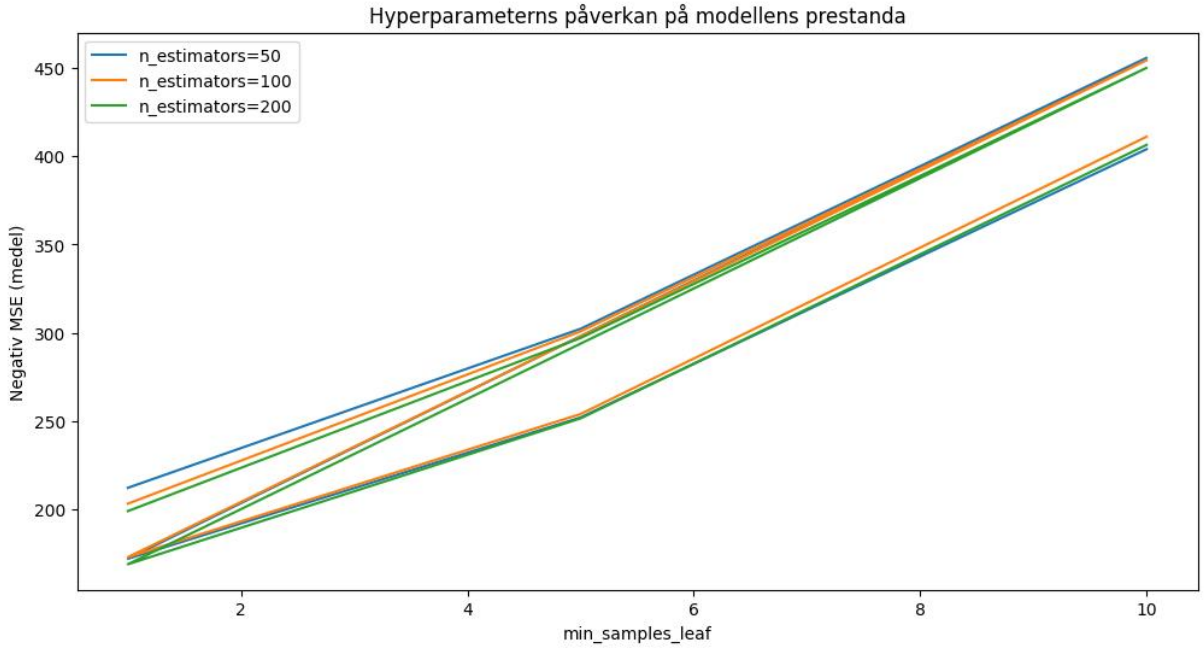
*Figur 2: **Beslutsdiagaram** (Exempel av Enskilt Beslutsträd)*



*Figur 3**:** Stapeldiagram – ”Feature Importance” (Visar hur viktiga varje funktion är för modellen.)*



Figur 4: Linjegraf – "Hyperparameter-tuning" (Visar hur olika värden på n_estimators och min_samples_leaf påverkar modellens prestanda i form av Mean Squared Error, MSE)



2.2.3 Gradient Boosting Regressor

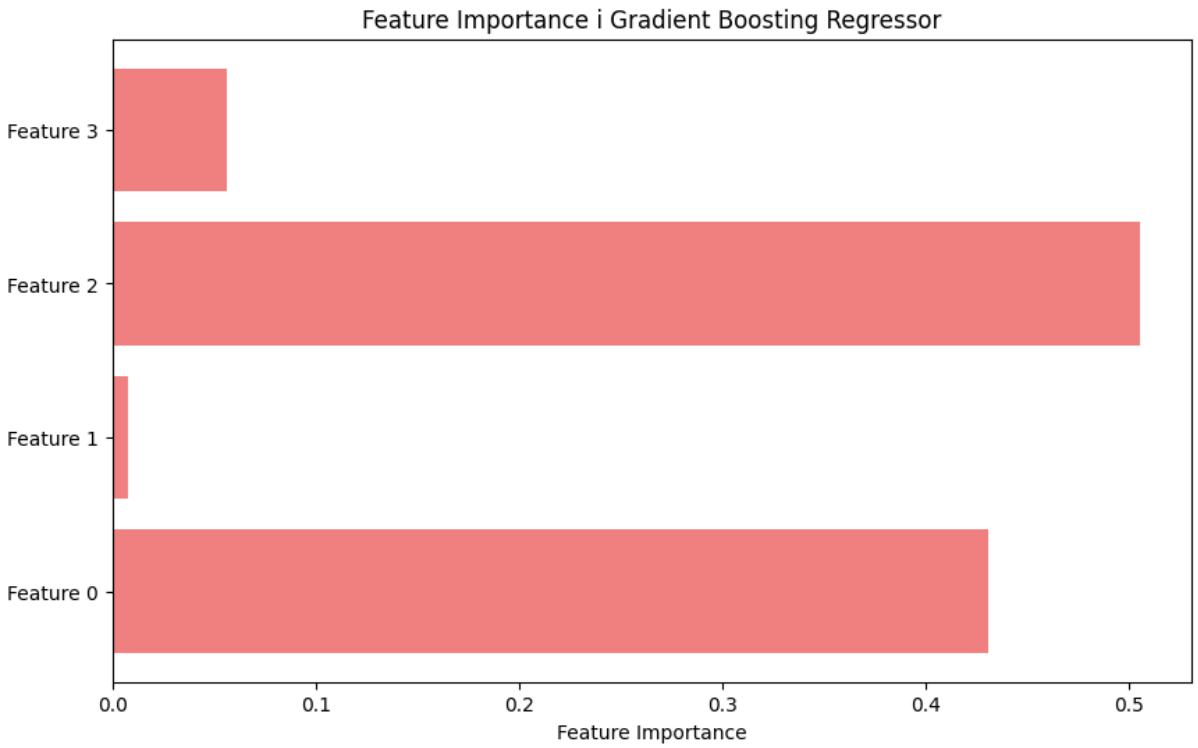
En sekventiell ensemblemodell som bygger träd där varje nytt träd korrigerar föregående fel.

2.2.3.1 Hyperparameter och Regularisering

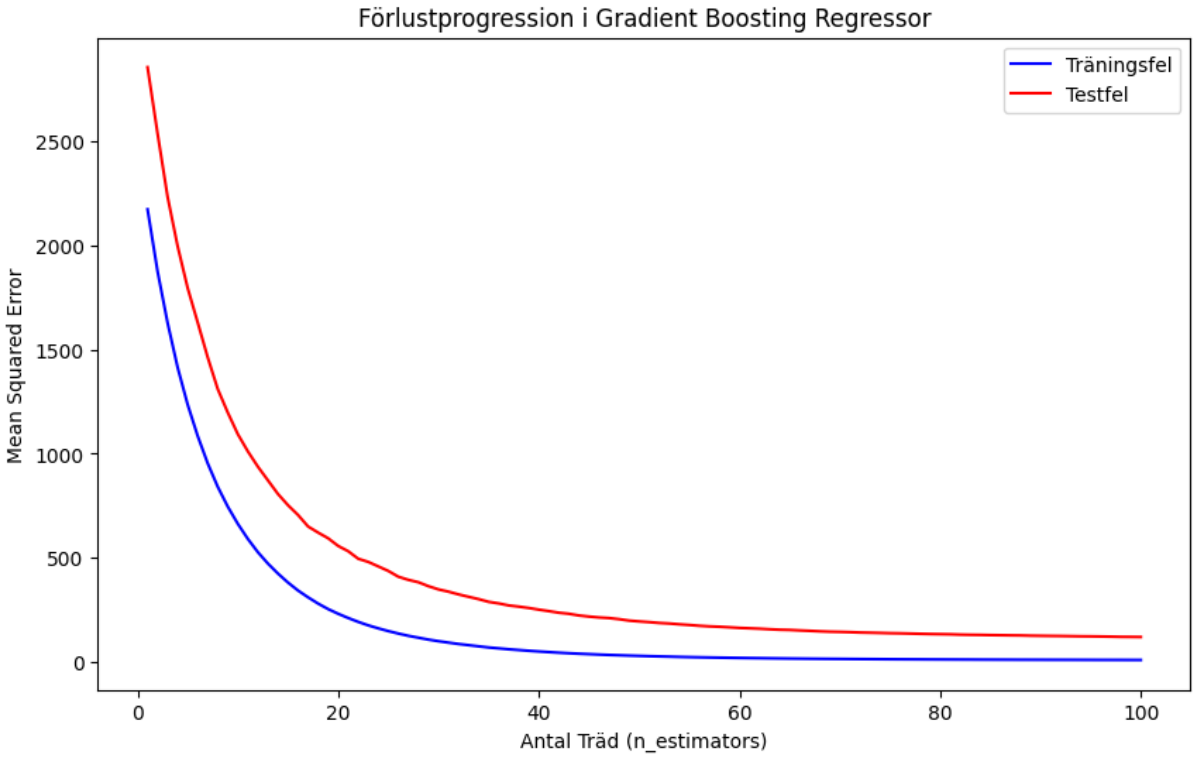
Gradient Boosting Regressor nyckelparametrar inkluderar:

- **n_estimators**: Antal träd.
- **learning_rate**: Vikt för varje nytt träd.
- **max_depth**: Maxdjup för varje träd för att undvika överanpassning.

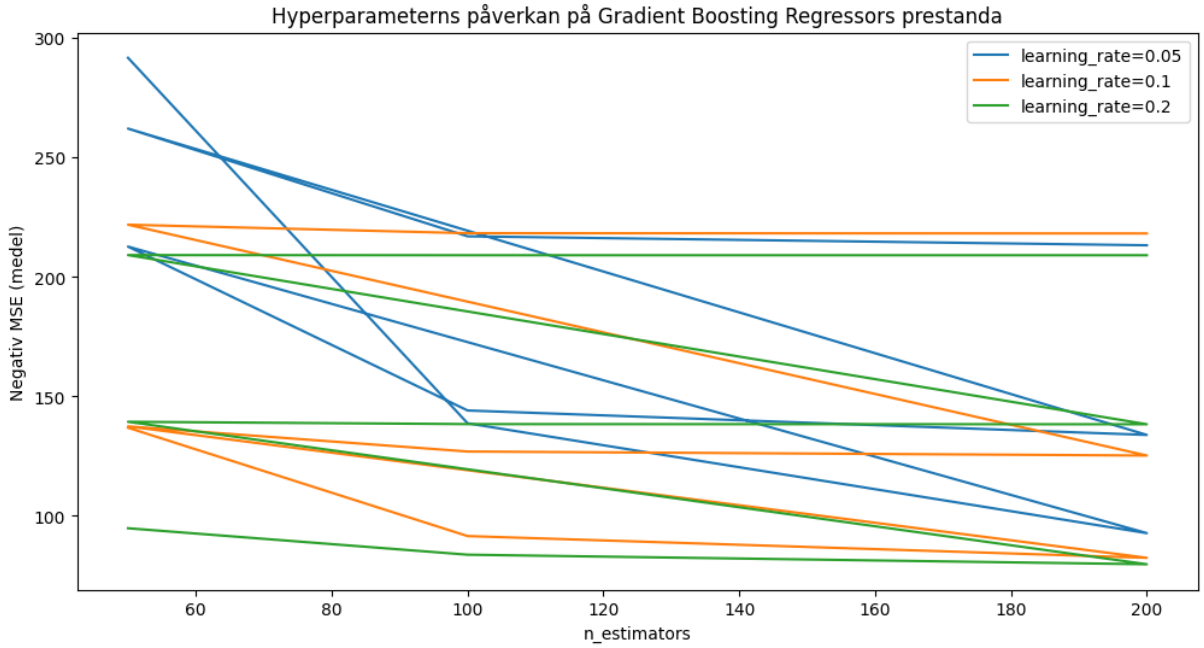
Figur 5: Stapeldiagram (”Feature Importance”)



Figur 6: Förlustprogressionsdiagram - Felvärden MSE på både tränings- och testdata för varje steg i iterationen. En bra modell bör minska felet gradvis, och vi kan se hur antalet träd påverkar prestandan.



Figur 7: Hyperparameter-justering Diagram – Identifieras de kombinationer av hyperparametrar som ger bäst prestanda.



2.2.4 Voting Regressor

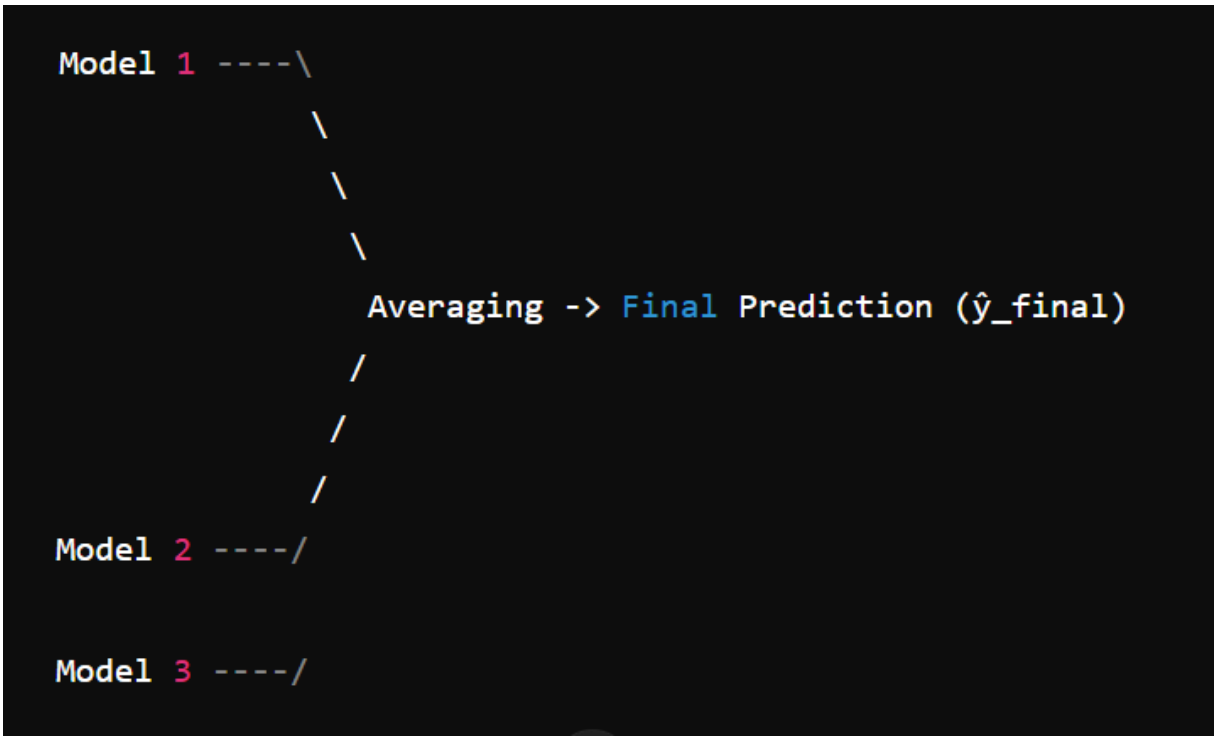
En ensemble där varje modell har en lika röst för det slutliga prediktionsvärdet. Den slutliga prediktionen för en Voting Regressor beräknas som medelvärdet av varje enskild modells prediktion i ensemblen. Till exempel, om vi har $\boldsymbol{n}$ modeller, blir formeln for slutprediktionen $\widehat{\boldsymbol{y}}\boldsymbol{final}$:

$$\widehat{\boldsymbol{y}}\boldsymbol{final} = \frac{1}{n}\sum_{i=1}^{n} \hat{y}_i$$

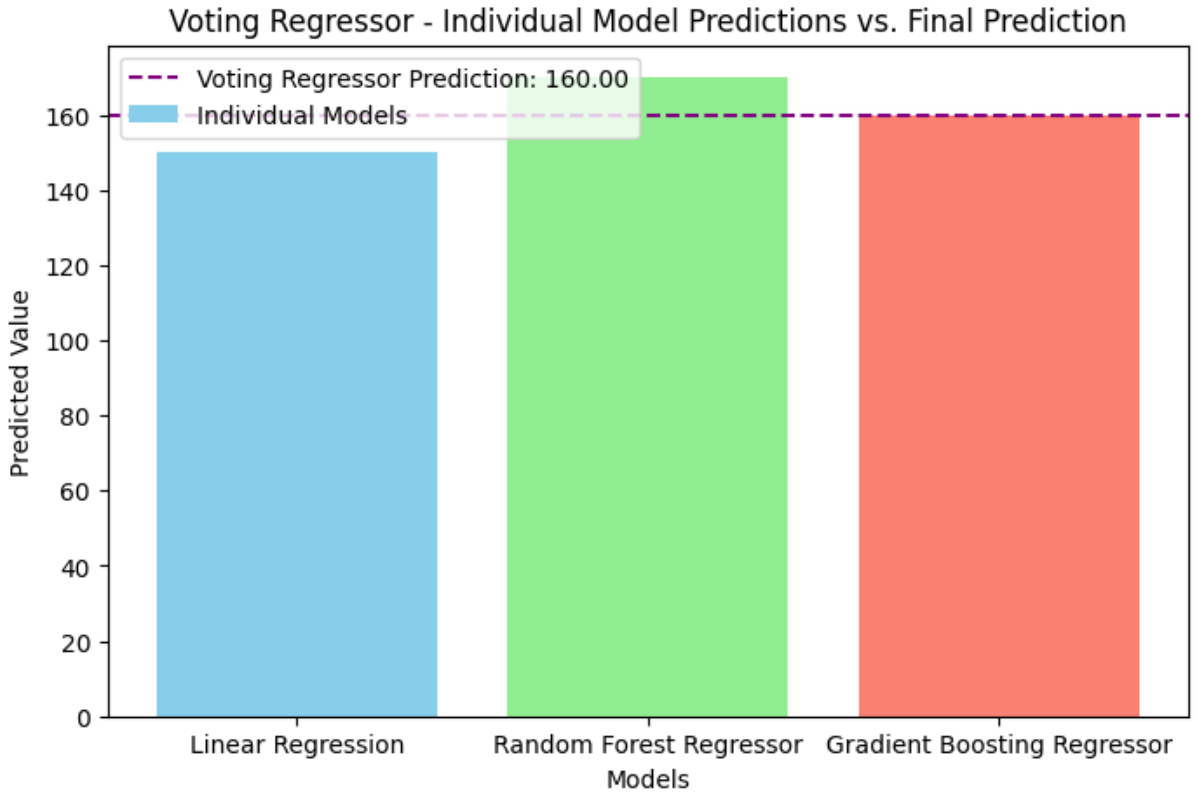
där:

- $\widehat{\boldsymbol{y}}\boldsymbol{final}$ är den slutliga prediktionen,
- $\boldsymbol{n}$ är antalet modeller i ensemblen,
- $\hat{\boldsymbol{y}}_{\boldsymbol{i}}$ är prediktionen från modell $\boldsymbol{i}$.

Figur 8: Enkelt Ensemblemodellers Diagram



Figur 9: Ensemblemodellers – Voting Regressor med Linear Regression, Random Forest Regressor, Gradient Boosting Regressor



2.3 Neurala Nätverk

Neurala nätverk är modeller inspirerade av hjärnan, med lager av noder som bearbetar information. Genom att justera vikter mellan noder lär de sig mönster och används för komplexa uppgifter som tidsserieförutsägelser.

2.3.1 LSTM (Long Short-Term Memory)

LSTM är en typ av artificiellt neuralt nätverk och en variant av ett rekurrent neuralt nätverk (RNN). Den är utvecklat för att hantera sekventiell data och lösa problem som kräver långsiktigt minne, såsom tidsserieförutsägelser, taligenkänning och naturlig språkbehandling.

2.3.1.1 Hyperparameter och Regularisering

Vid arbete med LSTM-nätverk är hyperparametrar och regularisering avgörande för prestanda och generalisering. Här är de viktigaste:

Hyperparametrar:

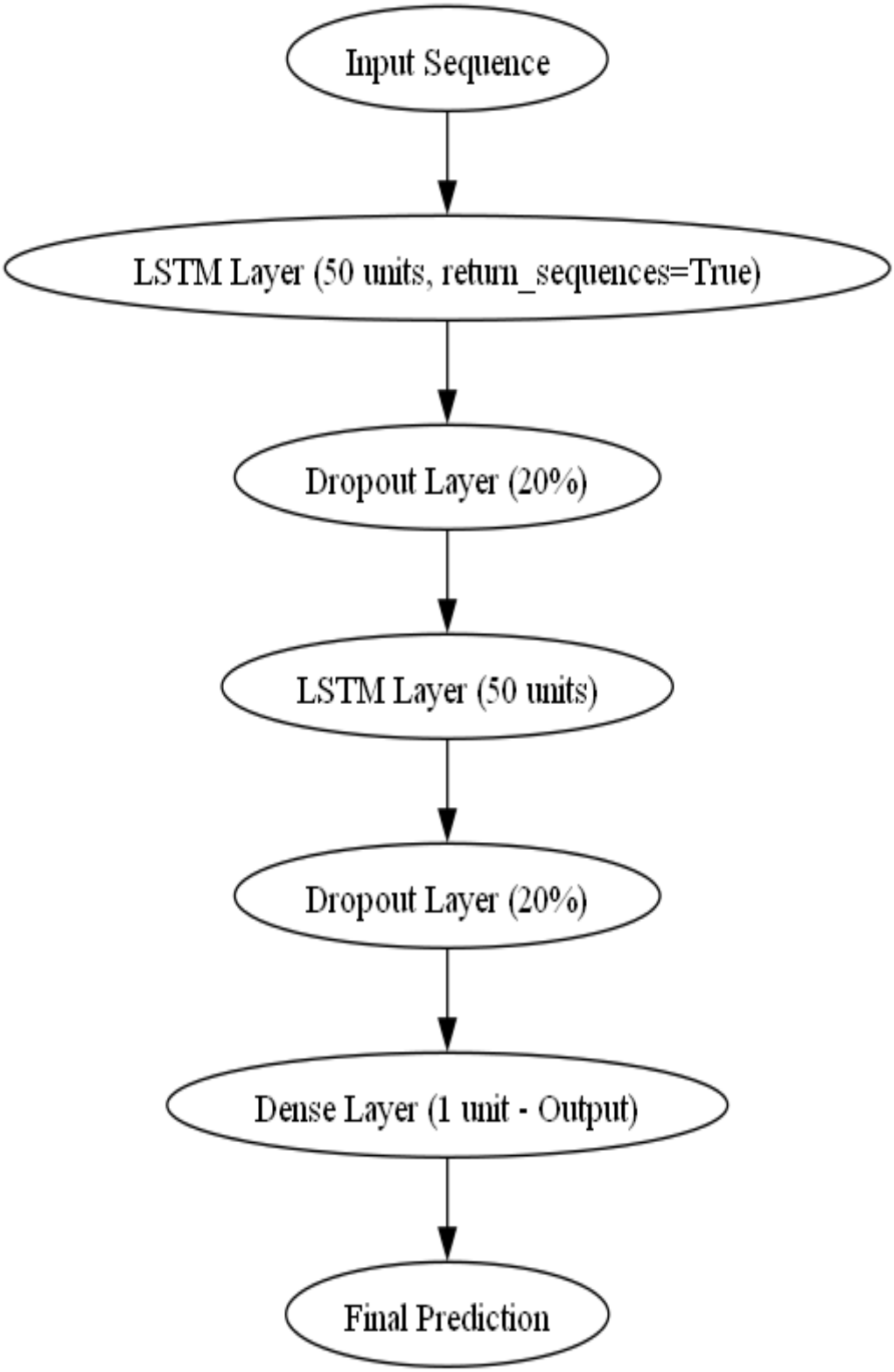
- Antal neuroner: Bestämmer varje lagers kapacitet att lära.
- Antal lager: Fler lager kan lära komplexa mönster men kan leda till överanpassning.
- Sekvenslängd: Antal tidssteg att beakta, vilket påverkar beräkningstid och mönsterigenkänning.
- Inlärningshastighet: Påverkar hur snabbt modellen anpassas.
- Batchstorlek och epoker: Påverkar träningsstabilitet och hastighet.

Regularisering:

- Dropout: Slumpar bort neuroner under träning för att minska överanpassning.
- Early Stopping: Stoppar träning om valideringsprestanda stagnerar.
- L1/L2-regularisering: Minskar stora vikter för bättre generalisering.
- Gradient Clipping: Begränsar gradienternas storlek för stabil träning.

Att justera dessa faktorer kan förbättra LSTM-modellens balans mellan inlärning och generalisering.

Figur 10: Diagramstruktur för en LSTM-modell



3 Metod

Arbetet genomfördes i flera steg, där data samlades in, förbereddes och analyserades för att skapa prognoser och uppskattningar för utbildningskostnader och befolkningstillväxt.

3.1 Datainsamling

För att genomföra arbetet har relevant data samlats in från offentliga källor och databaser. Datan har erhållits från följande steg:

1. Källor: Data om befolkning, födelsetal, dödsfall och utbildningskostnader har hämtats från SCB (Statistiska centralbyrån) samt från interna datalager (exempelvis kommunala databaser).
2. Datapreparering: Data har rensats och organiserats för att säkerställa konsekvens och kvalitet. Detta inkluderade att hantera saknade värden, standardisera format och skapa lämpliga tidsserier.
3. Databas: Data har lagrats i en SQLite-databas för att underlätta åtkomst och analys i Python. Tabellen innehåller information om regioner, år, åldersgrupper och kostnader per barn för både grundskola och gymnasieskola.
4. Uppdatering och validering: Regelbundet uppdaterad data har kontrollerats för att säkerställa att de senaste tillgängliga uppgifterna används. Datan har validerats mot externa källor och statistik för att säkerställa korrekthet.

Dessa steg har möjliggjort en robust databas för vidare analys och modellering.

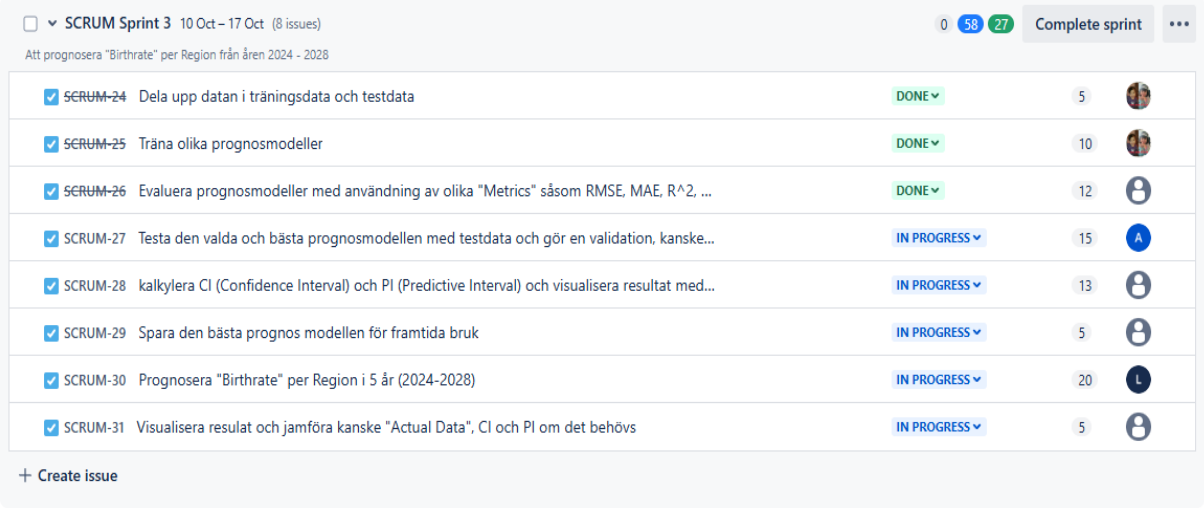
3.2 Agil arbetsmetodik

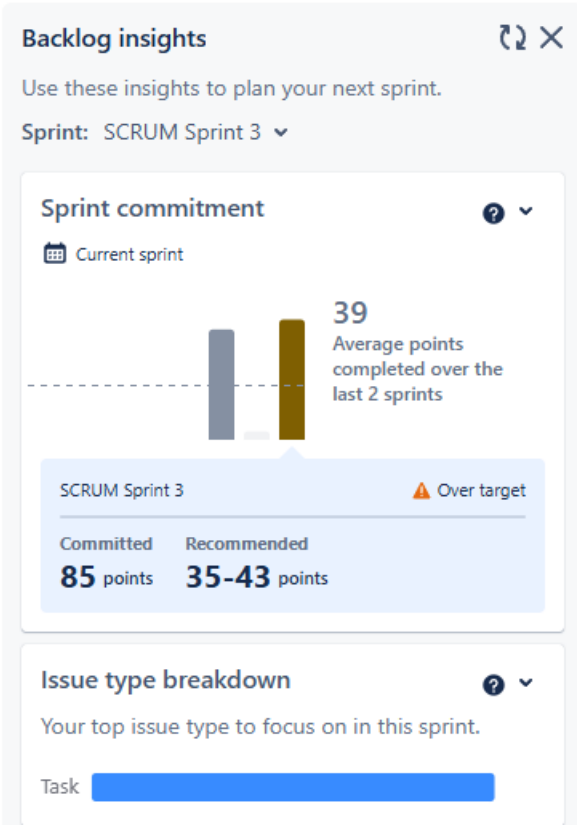
I detta projekt har jag initialt använt Jira som ett verktyg för att implementera agil arbetsmetodik. Målet var att organisera arbetsuppgifter och projektfaser i sprintar, vilket skulle möjliggöra en flexibel och iterativ arbetsprocess. Genom att skapa användarberättelser, definiera arbetsuppgifter och hålla regelbundna avstämningar planerade vi att upprätthålla en kontinuerlig framdrift och snabbt kunna anpassa oss till förändringar eller nya insikter.

Efter de första veckorna beslutade gruppen att övergå till en annan arbetsmetod, då det inte fanns ett gemensamt intresse för att fortsätta med Jira. Istället valde vi att arbeta mer informellt och fokusera på löpande kommunikation, men ändå med en agil grundsyn, där vi prioriterade flexibilitet och snabb anpassning till förändrade behov. Vi har fortsatt att tillämpa grundprinciperna från agil metodik, såsom samarbete, snabb återkoppling och fokus på värdeleverans, men utan det formella ramverket i Jira.

Sammanfattningsvis har vi anpassat oss efter behov och fortfarande haft ett agilt förhållningssätt, men utan att strikt följa alla steg i den agila processen. Nedan är skärmdumpar till vårt Jira:

Figur 11: Jira Skärmdumpar





4 Resultat och Diskussion

I detta avsnitt presenteras och diskuteras resultaten från prognoserna för både födelsetal och utbildningskostnader.

4.1 Resultat för Födelsetalsprognoser

I födelsetalsmodellen genomfördes en korsvalidering med fem foldar. Modellens prestanda mättes genom rot medelkvadratroten av felen (RMSE) för varje fold:

- **Fold 1**: RMSE = 0.0179
- **Fold 2**: RMSE = 0.0247
- **Fold 3**: RMSE = 0.0067
- **Fold 4**: RMSE = 0.0079
- **Fold 5**: RMSE = 0.0246

Den lägsta RMSE, 0.0067, uppnåddes i Fold 3, vilket indikerar att denna fold levererade den bästa modellen för födelsetalsprognosen. Efter att ha tränat modellen och valt den bästa, utfördes en stegvis tidsserieförutsägelse för de kommande 11 åren, vilket resulterade i följande värden (avrundade):

- År 1 (**2024**): 2756.3
- År 2 (**2025**): 2828.2
- År 3 (**2026**): 2901.7
- År 4 (**2027**): 2977.0
- År 5 (**2028**): 3052.5
- År 6 (**2029**): 3128.1
- År 7 (**2030**): 3203.7
- År 8 (**2031**): 3279.4
- År 9 (**2032**): 3355.1
- År 10 (**2033**): 3430.8
- År 11 (**2034**): 3506.6

Dessa prognosvärden visar en gradvis ökning av födelsetalen, vilket antyder en potentiell befolkningstillväxt de närmaste åren. Modellen verkar tillförlitlig för prognoser på kort sikt, baserat på låga RMSE-värden och stabila prediktioner.

4.2 Resultat för Utbildningskostnadsprognoser

Utbildningskostnadsmodellen analyserades för både fasta och nuvarande kostnader för grundskola och gymnasieskola utbildning. Prestandan för de olika kostnadskategorierna utvärderades med hjälp av RMSE, MAE och R^2:

a. **Fasta kostnader för grundskola**:
   - RMSE: 2640.46
   - MAE: 2257.67
   - R^2: 0.9581 = **96%**
b. **Fasta kostnader för gymnasieskola**:
   - RMSE: 2137.33
   - MAE: 1813.09
   - R^2: 0.9596 = **96%**
c. **Nuvarande kostnader för grundskola**:
   - RMSE: 2480.15
   - MAE: 2112.47
   - R^2: 0.9794 = **98%**
d. **Nuvarande kostnader för gymnasieskola**:
   - RMSE: 2992.93
   - MAE: 2681.99
   - R^2: 0.9642 = **96%**

Höga R^2-värden för samtliga kostnadskategorier indikerar att modellerna kan förklara en betydande andel av variansen i utbildningskostnaderna. Särskilt för rörliga kostnader för primär utbildning, där R^2 var 0.9794, uppnåddes mycket god precision. De lägre MAE- och RMSE-värdena visar också att modellerna har relativt små avvikelser från de faktiska värdena, vilket gör prognoserna tillförlitliga för både budgetering och ekonomisk planering.

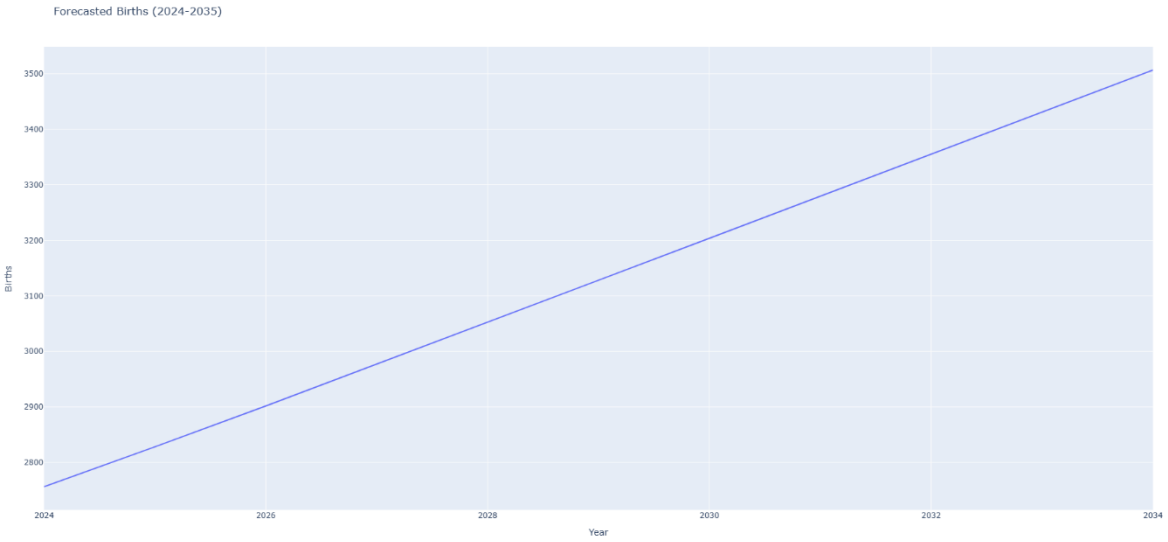
4.3 Diskussion

Resultaten visar att de implementerade modellerna för födelsetals- och kostnadsprognoser är både robusta och noggranna. De relativt låga RMSE- och MAE-värdena, tillsammans med höga R^2, indikerar god modellpassning och låg prognososäkerhet. Dessa modeller kan användas som verktyg för att förutsäga framtida behov och kostnader inom primär och sekundär utbildning, vilket kan underlätta budgetering och resursplanering i kommuner.

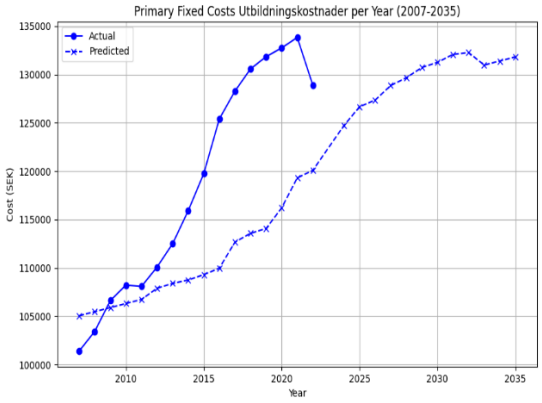
En potentiell förbättring är att inkludera migration och regionala socioekonomiska faktorer i prognosmodellen för att ytterligare minska osäkerheten i resultaten. Dessutom kan fortsatt

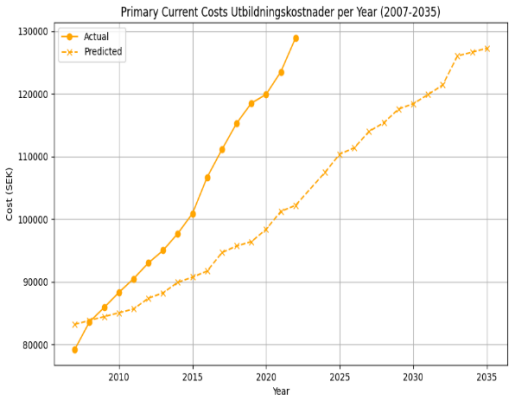
utvärdering och uppdatering av modellen med nya data bidra till att bibehålla prognosernas precision över tid.

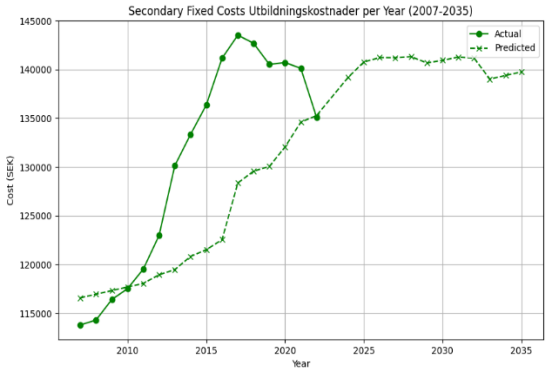
Figur 12: Visualisering av Födelsetal Prognos (2024–2035)

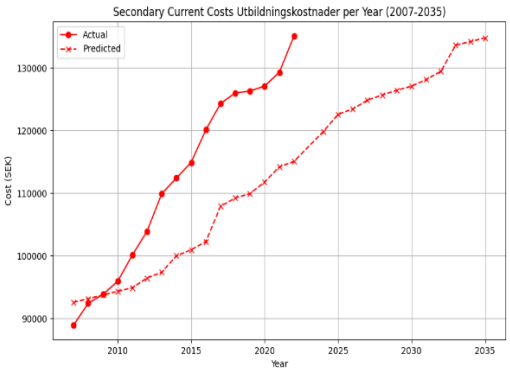


Figur 13: Visualisering av ”Actual vs. Predicted” Utbildningskostnader

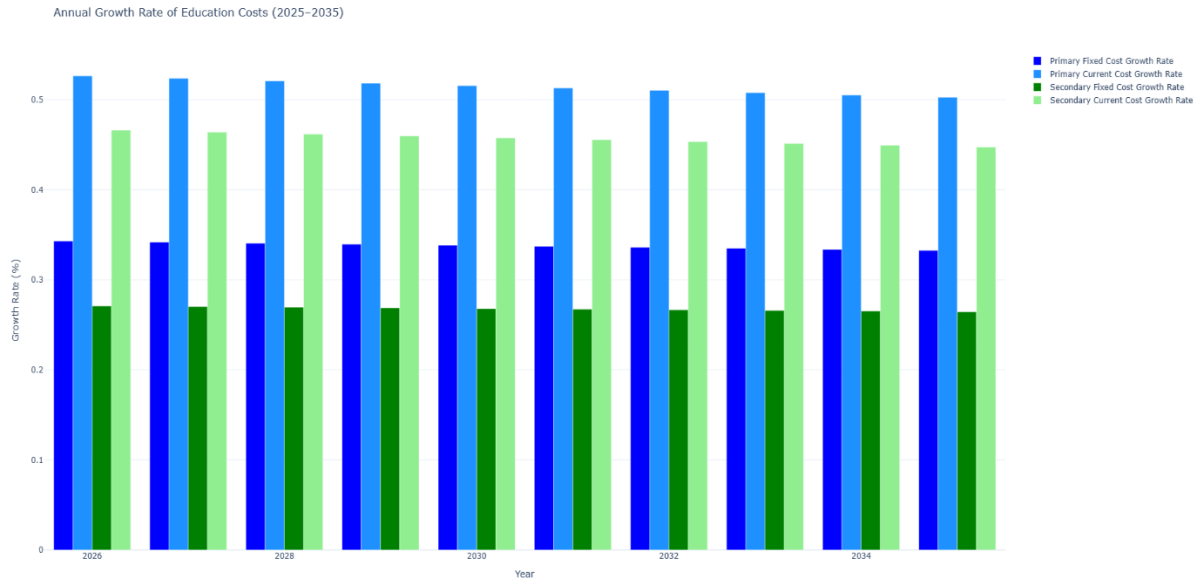








Figur 14: Visualisering av "Annual Growth Rate of Education Costs" (2025-2035)



5 Slutsatser

Påverkansfaktorer på utbildningskostnader: De huvudsakliga faktorerna som påverkar utbildningskostnaderna i olika regioner i Sverige inkluderar förändringar i befolkningsstrukturen, såsom antal barn och ungdomar i skolåldern, regionalekonomisk utveckling samt inflation och kostnader för personal och resurser. Dessa variabler varierar mellan regioner, vilket bidrar till skillnader i kostnadsutveckling för utbildningssystemet.

Födelsetalsprognos och befolkningsstrukturens inverkan på utbildningskostnader: Prognoser visar en gradvis ökning av födelsetalen de kommande 11 åren, från 2756.3 år 2024 till 3506.6 år 2034. Den lägsta RMSE-värdet (0.0067) uppnåddes i fold 3, vilket bekräftar modellens noggrannhet för kortsiktiga prognoser. Ökande födelsetal förväntas öka antalet barn i skolåldern, vilket direkt påverkar framtida utbildningskostnader. Utbildningskostnadsmodellen för både fasta och nuvarande kostnader visade höga R^2-värden, särskilt för rörliga kostnader i grundskolan (R^2 = 0.9794), vilket indikerar att modellerna är tillförlitliga för att förutsäga budgetbehov.

Betydelse för planering och budgetering: De låga RMSE- och MAE-värdena samt höga R^2-värdena bekräftar att modellerna har hög precision och är väl lämpade för att förutsäga framtida utbildningskostnader och befolkningsbehov. Dessa insikter kan hjälpa beslutsfattare att optimera budgeteringen och effektivt fördela resurser inom utbildningssektorn utifrån framtida behov och regionala skillnader.

Framtida förbättringar: Att inkludera migration och socioekonomiska faktorer kan ytterligare förbättra prognosernas noggrannhet. Fortsatt modellutvärdering och uppdatering med nya data rekommenderas för att bibehålla och förbättra prognosernas precision över tid.

6 Självutvärdering

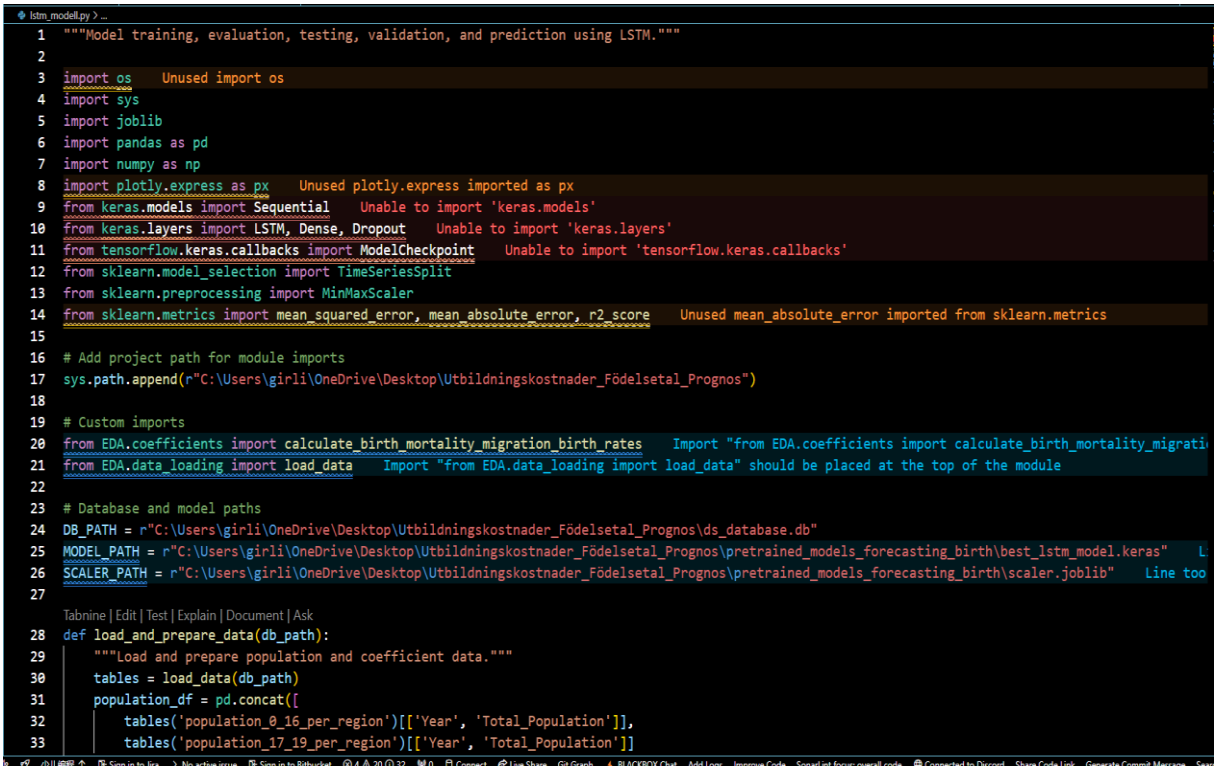
Utmaningarna inkluderade hantering av stora datamängder och val av lämpliga modeller för exakta prognoser. Genom systematisk felsökning, användning av robusta valideringsmetoder och justeringar i modellen hanterades dessa effektivt för att optimera resultatens tillförlitlighet.

Jag anser att jag förtjänar betyget G, då jag tyvärr var sen med att lämna in mitt projekt. Trots detta har jag uppfyllt projektets krav, tagit hänsyn till detaljer i analysen och levererat tillförlitliga prognoser som kan användas för resursplanering.

Tack Antonio och Linus för stödet och vägledningen under projektet. Din feedback har varit avgörande för att stärka analysens kvalitet och riktighet.

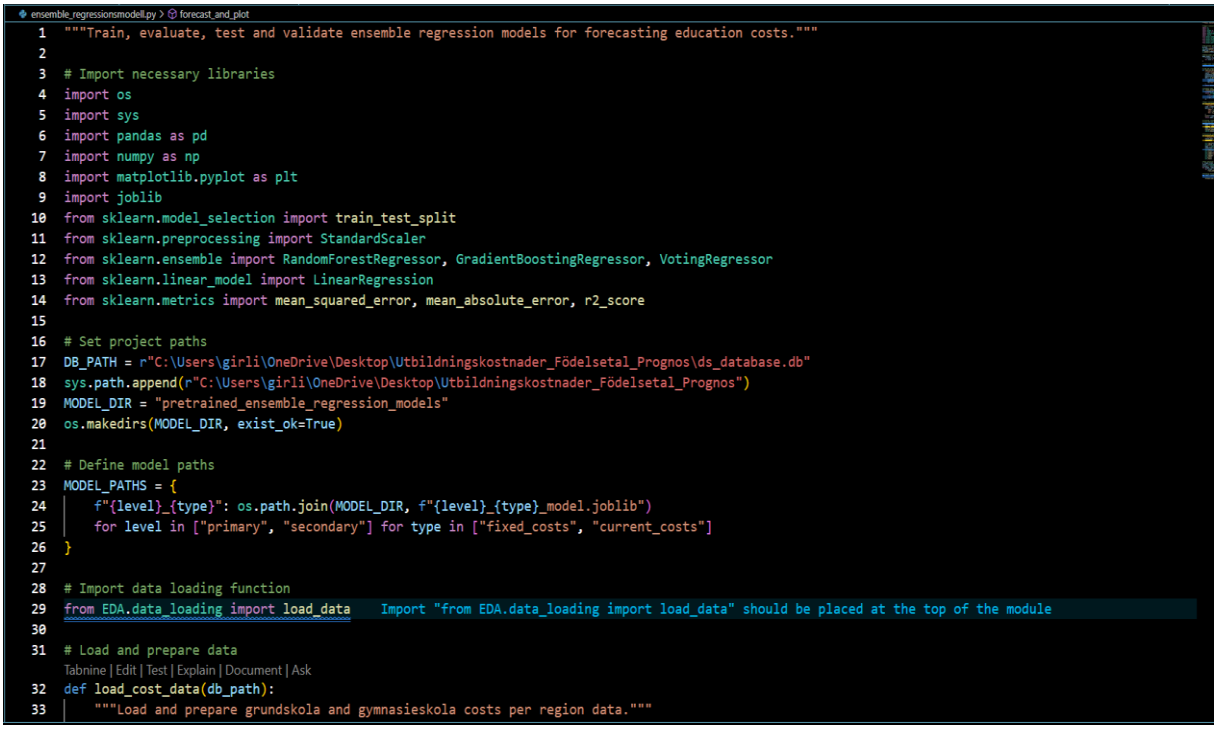
Appendix A

Code Snippets: **”lstm_modell.py”**



```python
"""Model training, evaluation, testing, validation, and prediction using LSTM."""

import os    Unused import os
import sys
import joblib
import pandas as pd
import numpy as np
import plotly.express as px    Unused plotly.express imported as px
from keras.models import Sequential    Unable to import 'keras.models'
from keras.layers import LSTM, Dense, Dropout    Unable to import 'keras.layers'
from tensorflow.keras.callbacks import ModelCheckpoint    Unable to import 'tensorflow.keras.callbacks'
from sklearn.model_selection import TimeSeriesSplit
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score    Unused mean_absolute_error imported from sklearn.metrics

# Add project path for module imports
sys.path.append(r"C:\Users\girli\OneDrive\Desktop\Utbildningskostnader_Födelsetal_Prognos")

# Custom imports
from EDA.coefficients import calculate_birth_mortality_migration_birth_rates    Import "from EDA.coefficients import calculate_birth_mortality_migrati
from EDA.data_loading import load_data    Import "from EDA.data_loading import load_data" should be placed at the top of the module

# Database and model paths
DB_PATH = r"C:\Users\girli\OneDrive\Desktop\Utbildningskostnader_Födelsetal_Prognos\ds_database.db"
MODEL_PATH = r"C:\Users\girli\OneDrive\Desktop\Utbildningskostnader_Födelsetal_Prognos\pretrained_models_forecasting_birth\best_lstm_model.keras"    L
SCALER_PATH = r"C:\Users\girli\OneDrive\Desktop\Utbildningskostnader_Födelsetal_Prognos\pretrained_models_forecasting_birth\scaler.joblib"    Line too

Tabnine | Edit | Test | Explain | Document | Ask
def load_and_prepare_data(db_path):
    """Load and prepare population and coefficient data."""
    tables = load_data(db_path)
    population_df = pd.concat([
        tables('population_0_16_per_region')[['Year', 'Total_Population']],
        tables('population_17_19_per_region')[['Year', 'Total_Population']]
```

Code Snippets: **“ensemble_regressionsmodeller.py”**

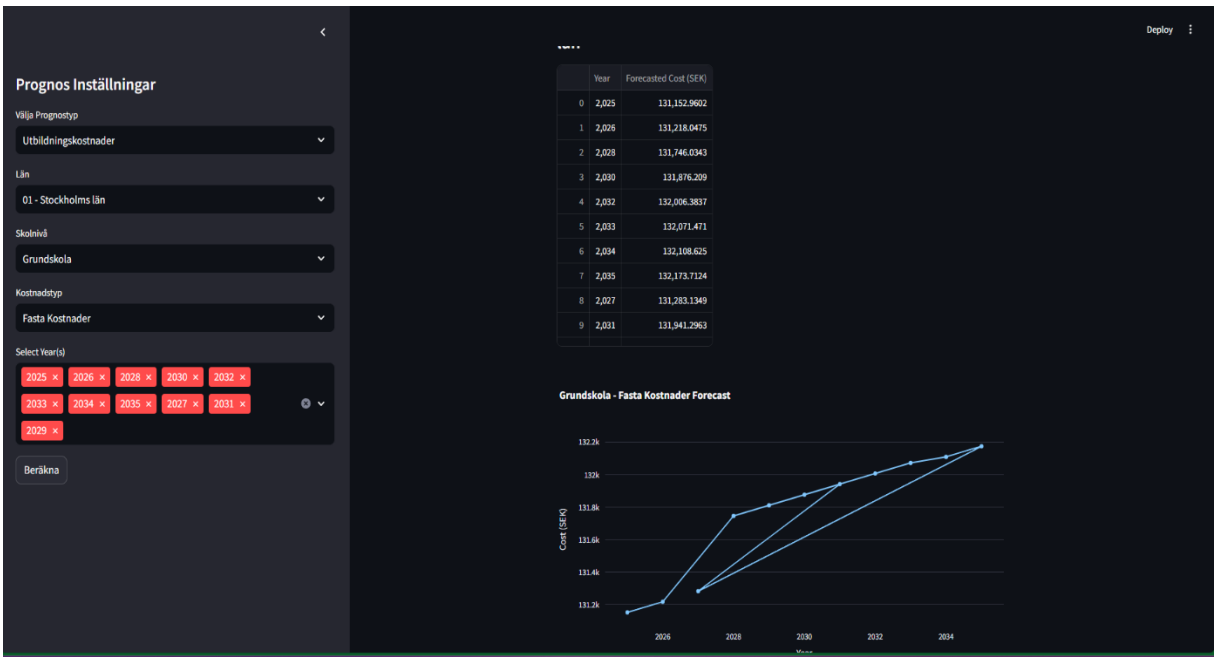


```python
"""Train, evaluate, test and validate ensemble regression models for forecasting education costs."""

# Import necessary libraries
import os
import sys
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import joblib
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor, VotingRegressor
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score

# Set project paths
DB_PATH = r"C:\Users\girli\OneDrive\Desktop\Utbildningskostnader_Födelsetal_Prognos\ds_database.db"
sys.path.append(r"C:\Users\girli\OneDrive\Desktop\Utbildningskostnader_Födelsetal_Prognos")
MODEL_DIR = "pretrained_ensemble_regression_models"
os.makedirs(MODEL_DIR, exist_ok=True)

# Define model paths
MODEL_PATHS = {
    f"{level}_{type}": os.path.join(MODEL_DIR, f"{level}_{type}_model.joblib")
    for level in ["primary", "secondary"] for type in ["fixed_costs", "current_costs"]
}

# Import data loading function
from EDA.data_loading import load_data    Import "from EDA.data_loading import load_data" should be placed at the top of the module

# Load and prepare data
Tabnine | Edit | Test | Explain | Document | Ask
def load_cost_data(db_path):
    """Load and prepare grundskola and gymnasieskola costs per region data."""
```

Code Snippets: **“streamlit (app.py)”**

```python
"""Streamlit application for forecasting education costs and birth rates in Sweden."""
import streamlit as st
import joblib
import numpy as np
import pandas as pd
import plotly.graph_objects as go
from tensorflow.keras.models import load_model    Unable to import 'tensorflow.keras.models'
from sklearn.preprocessing import MinMaxScaler

# Paths to pretrained models
MODEL_PATHS = {
    "primary_fixed_costs": r"C:\Users\girli\OneDrive\Desktop\Utbildningskostnader_Födelsetal_Prognos\pretrained_ensemble_regression_models\primary_fix
    "primary_current_costs": r"C:\Users\girli\OneDrive\Desktop\Utbildningskostnader_Födelsetal_Prognos\pretrained_ensemble_regression_models\primary_c
    "secondary_fixed_costs": r"C:\Users\girli\OneDrive\Desktop\Utbildningskostnader_Födelsetal_Prognos\pretrained_ensemble_regression_models\secondary_
    "secondary_current_costs": r"C:\Users\girli\OneDrive\Desktop\Utbildningskostnader_Födelsetal_Prognos\pretrained_ensemble_regression_models\seconda
    "birth_forecast": r"C:\Users\girli\OneDrive\Desktop\Utbildningskostnader_Födelsetal_Prognos\pretrained_models_forecasting_birth\best_lstm_model.ke
    "scaler": r"C:\Users\girli\OneDrive\Desktop\Utbildningskostnader_Födelsetal_Prognos\pretrained_models_forecasting_birth\scaler.joblib"    Line too
}

# Load models
education_models = {key: joblib.load(path) for key, path in MODEL_PATHS.items() if 'costs' in key}
birth_model = load_model(MODEL_PATHS['birth_forecast'])
scaler = joblib.load(MODEL_PATHS['scaler'])

# Region mapping
region_mapping = {
    "01": "Stockholms län", "03": "Uppsala län", "04": "Södermanlands län", "05": "Östergötlands län",    Line too long (102/100)
    "06": "Jönköpings län", "07": "Kronobergs län", "08": "Kalmar län", "09": "Gotlands län",
    "10": "Blekinge län", "12": "Skåne län", "13": "Hallands län", "14": "Västra Götalands län",
    "17": "Värmlands län", "18": "Örebro län", "19": "Västmanlands län", "20": "Dalarnas län",
    "21": "Gävleborgs län", "22": "Västernorrlands län", "23": "Jämtlands län", "24": "Västerbottens län",    Line too long (106/100)
    "25": "Norrbottens län"
}
```

Skärmdump: **Streamlit app**



Källförteckning

SCB Statistiskdatabasen. (2023). *Antal barn per region i åldrarna 0 - 16 år från 1968 - 2023*. Retrieved from SCB Statistisksdatabas: https://www.statistikdatabasen.scb.se/pxweb/sv/ssd/START__BE__BE0101__BE0101A/BefolkningNy

SCB Statistisksdatabasen. (2022). *Grundskola och Gymnasieskola fasta och löpande kostnader från år 2007 - 2022*. Retrieved from SCB Statistisksdatabas: https://www.statistikdatabasen.scb.se/pxweb/sv/ssd/START__UF__UF0514/UtbKostGrundGym

SCB Statistisksdatabasen. (2023). *Antal avlidna barn per region år 2000 - 2023*. Retrieved from SCB Statistisksdatabas: https://www.statistikdatabasen.scb.se/pxweb/sv/ssd/START__BE__BE0101__BE0101I/DodaFodelsearK/

SCB Statistisksdatabasen. (2023). *Antal barn per region i åldrarna 17 – 19 år från 1968 - 2023*. Retrieved from SCB Statistisksdatabas: https://www.statistikdatabasen.scb.se/pxweb/sv/ssd/START__BE__BE0101__BE0101A/BefolkningNy

SCB Statistisksdatabasen. (2023). *Flyttningar efter region och ålder år 1997 - 2023*. Retrieved from SCB Statistisksdatabas: https://www.statistikdatabasen.scb.se/pxweb/sv/ssd/START__BE__BE0101__BE0101J/Flyttningar97/

SCB Statistisksdatabasen. (2023). *Födda efter region år 1968 - 2023*. Retrieved from SCB Statistisksdatabas: https://www.statistikdatabasen.scb.se/pxweb/sv/ssd/START__BE__BE0101__BE0101H/FoddaK/

Scikit-learn. (n.d.). *Plot individual and voting regression predictions*. Retrieved from Scikit-learn: https://machinelearningmastery.com/how-to-develop-lstm-models-for-time-series-forecasting

Brownlee, J. (2020, August 28). *How to Develop LSTM Models for Time Series Forecasting*. Retrieved from Machine Learning Mastery: https://machinelearningmastery.com/how-to-develop-lstm-models-for-time-series-forecasting