

SOK-1006-Mappe2

83

Kapittel 1.

Hva er en bompengavgift, og hvorfor er den blitt innført?

5. Januar 2023 ble det innført bompengavgift i Tromsø, det fungerer i praksis slik at når du i din privatbil passerer et bompunge punkt vil du i etterkant bli belastet et visst beløp. I tillegg har du i Tromsø i dag muligheten å velge mellom 2 avtaler, disse heter bypakke og byvekstavgift, og bestemmer hva beløpet du betaler bidrar med i samfunnet.

Byvekstavgift er tanken om at veksten i trafikk i storbyer ikke skal fortsette som personbilvekst, men som kollektivtrafikk, delvis myke trafikanter og gange. Derfor blir det vanligere å sette en avgift på personbiler som bruker tungt trafikkerte veier i større byer over hele Norge.

Bompengeløsningen er i bruk i flere byer i Norge, i Oslo fungerer den i likhet som i Tromsø, med fast pris per passering og diverse rabatter basert på når du kjører og hva du kjører med.

En viktig forskjell mellom å implementere bompenger som en avgift istedenfor å sette dette som en skatt er at bompenger kun vil påvirke bilister som kjører i de bompengebelastede områdene, på denne måten har vegvesenet laget en løsning hvor de høy belastede veiene finansierer seg selv. Ikke bare samler vegvesenet inn penger for vedlikehold, men det vil være et viktig verktøy framover for å regulere trafikken, som nevnt tidligere er et mål i seg selv.

Kapittel 2 Trafikkbildet i Tromsø før og etter innføring av bompenger

Til analysen er det valgt 3 veipasseringer inn til Tromsø, dette er Sandnessundbrua, Tromsøysundtunnelen og Tromsøbrua.

Det er i praksis ikke mulighet for å kjøre inn på Tromsø øya uten å betale bomavgift, det er bomavgifts sensor etter flyplassen, etter brua fra Tromsdalen og ved Tomasjord som forhindrer bilister å kjøre uten å bli detektert av disse sensorene. Om du belastes basert på skanning av registreringsnummeret ditt er det full uttelling på avgiften, dersom du har løsningen med AutoPASS vil den endelige belastningen kuttes med 20%, om du i tillegg til AutoPASS avtale har elbil vil man også ha muligheten til å få videre 40% kutt i denne avgiften, sammensatt har du mulighet til å få et avslag på 60% av denne avgiften.

Hvor mye du ender opp med å legge ut i denne utgiften vil også variere fra bilist til bilist, som en del av bybilde politikken er det satt mål å ikke bare endre på antall biler i trafikken på øya, men også hvilke type biler som er å finne i trafikken.

```
import pandas as pd
import numpy as np
import sympy as sp
from matplotlib import pyplot as plt
from sklearn.linear_model import LinearRegression
import markdown
```

```
df_22 = pd.read_csv("/azhome/jma059@ad.uit.no/Notebooks endr/notebooks/sok-1006/sok-1006/s
df_23 = pd.read_csv("/azhome/jma059@ad.uit.no/Notebooks endr/notebooks/sok-1006/sok-1006/s
```

```
veitrafikk22 = df_22.loc[df_22['Felt'] == 'Totalt']
veitrafikk23 = df_23.loc[df_23['Felt'] == 'Totalt']
```

```
veitrafikk22['Dato'] = pd.to_datetime(veitrafikk22['Dato'], format='%d.%m.%Y')
veitrafikk23['Dato'] = pd.to_datetime(veitrafikk23['Dato'], format='%d.%m.%Y')
```

```
veitrafikk22['Måned'] = veitrafikk22['Dato'].dt.month
veitrafikk23['Måned'] = veitrafikk23['Dato'].dt.month
```

```
/tmp/ipykernel_57371/846334806.py:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide

```
veitrafikk22['Dato'] = pd.to_datetime(veitrafikk22['Dato'], format='%d.%m.%Y')
```

```
/tmp/ipykernel_57371/846334806.py:8: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide

```
veitrafikk23['Dato'] = pd.to_datetime(veitrafikk23['Dato'], format='%d.%m.%Y')
```

```
/tmp/ipykernel_57371/846334806.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide

```
veitrafikk22['Måned'] = veitrafikk22['Dato'].dt.month
```

```
/tmp/ipykernel_57371/846334806.py:11: SettingWithCopyWarning:
```

A value is trying to be set on a copy of a slice from a DataFrame.
Try using `.loc[row_indexer,col_indexer] = value` instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide
`veitrafikk23['Måned'] = veitrafikk23['Dato'].dt.month`

```
total_veitrafikk22 = veitrafikk22.groupby('Måned', as_index=True)['Trafikkmengde'].sum().t  
total_veitrafikk23 = veitrafikk23.groupby('Måned', as_index=True)['Trafikkmengde'].sum().t  
total_veitrafikk22
```

| | Måned | Trafikkmengde |
|----|-------|---------------|
| 0 | 1 | 1107405 |
| 1 | 2 | 1101796 |
| 2 | 3 | 1261193 |
| 3 | 4 | 1156751 |
| 4 | 5 | 1309975 |
| 5 | 6 | 1338166 |
| 6 | 7 | 1069357 |
| 7 | 8 | 1288032 |
| 8 | 9 | 1305452 |
| 9 | 10 | 1268636 |
| 10 | 11 | 1222719 |
| 11 | 12 | 1187268 |

Viser den totale trafikkmengden i Tromsø fra Januar til Desember i 2022.

```
total_veitrafikk23
```

| | Måned | Trafikkmengde |
|---|-------|---------------|
| 0 | 1 | 1049468 |
| 1 | 2 | 993131 |
| 2 | 3 | 1129464 |
| 3 | 4 | 975127 |
| 4 | 5 | 695589 |

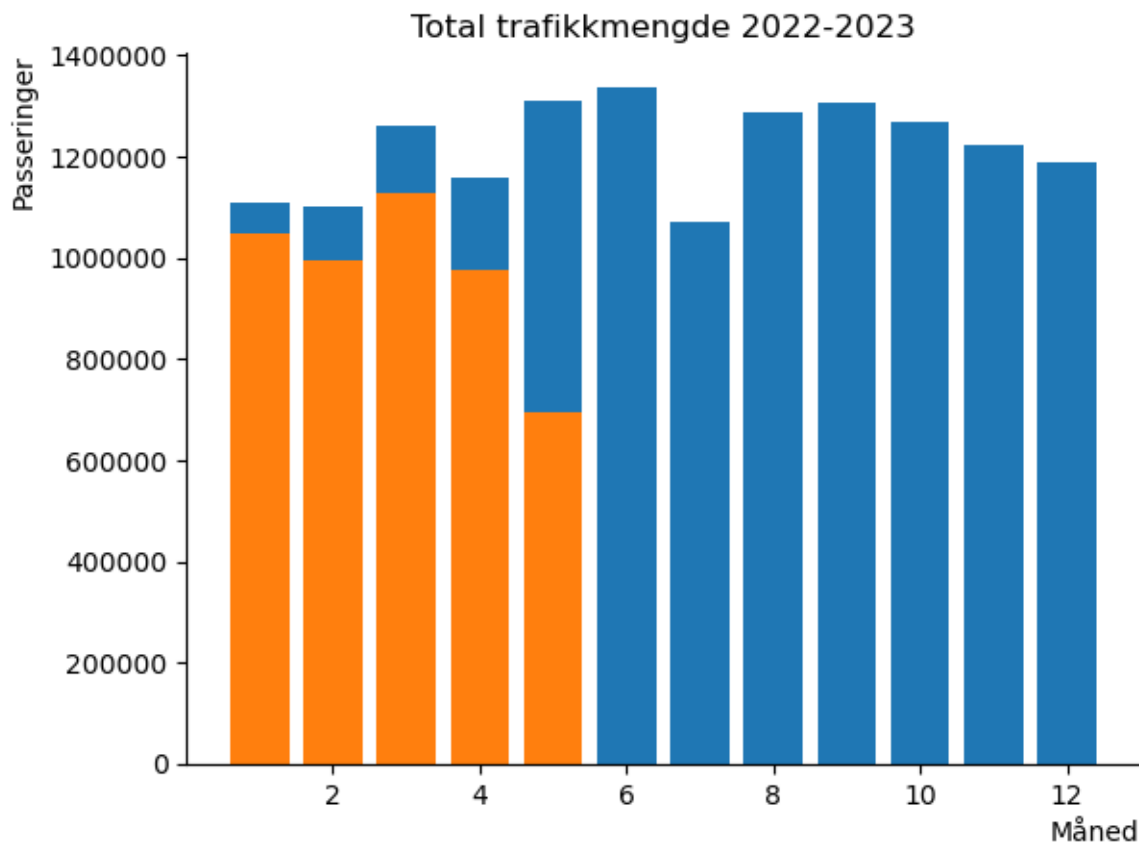
Viser den totale trafikkmengden i Tromsø fra Januar til Mai i 2023.

```
total_veitrafikk23['trafikkdifferanse'] = total_veitrafikk22['Trafikkmengde'] - total_veit  
total_veitrafikk23['trafikkdifferanse']
```

```
0      57937.0  
1     108665.0  
2     131729.0  
3     181624.0  
4     614386.0  
Name: trafikkdifferanse, dtype: float64
```

Viser differansen mellom den registrerte trafikken i 2023 og trafikken i 2022.

```
def create_ax():  
    fig, ax = plt.subplots()  
    ax.set_ylabel('Passeringer', loc='top')  
    ax.set_xlabel('Måned', loc='right')  
    ax.spines['top'].set_color('none')  
    ax.spines['right'].set_color('none')  
  
    return fig, ax  
  
fig, ax = create_ax()  
  
ax.bar(total_veitrafikk22['Måned'], total_veitrafikk22['Trafikkmengde'])  
ax.bar(total_veitrafikk23['Måned'], total_veitrafikk23['Trafikkmengde'])  
  
ax.set_title('Total trafikkmengde 2022-2023')  
  
ax.ticklabel_format(style='plain')
```



Vi observerer i denne figuren at sammenlignet med dataen i 2022, som er vist som de blå stolpene, har den totale personbiltrafikken gått ned etter innføringen av bompenger. Det ble etter 5. Januar registrert en nedgang på hele 11% etter innføringen av bompenger. De som trenger bil kan nok ikke gjøre seg en tilpassning til kollektivtrafikk over natten som vil foreslå at den første nedgangen bilister på veiene var i de som kjørte unødige.

```
veitrafikk23 = veitrafikk23.groupby('Dato', as_index=True)['Trafikkmengde'].sum().to_frame()
```

```
df_buss = pd.read_csv("https://raw.githubusercontent.com/uit-sok-1006-v23/uit-sok-1006-v23")
df_buss = df_buss.groupby('date', as_index=True)['Påstigende'].sum().to_frame().reset_index()
df_buss22 = df_buss[(df_buss['date'] >= '2022-01-01') & (df_buss['date'] <= '2022-04-30')]
df_buss23 = df_buss[(df_buss['date'] >= '2023-01-01') & (df_buss['date'] <= '2023-04-30')]
```

```
def create_ax():
    fig, ax = plt.subplots()
```

```

        ax.set_ylabel('Passeringer', loc='top')
        ax.set_xlabel('Måned', loc='right')
        ax.spines['top'].set_color('none')
        ax.spines['right'].set_color('none')
        plt.xticks(color='w')

    return fig, ax

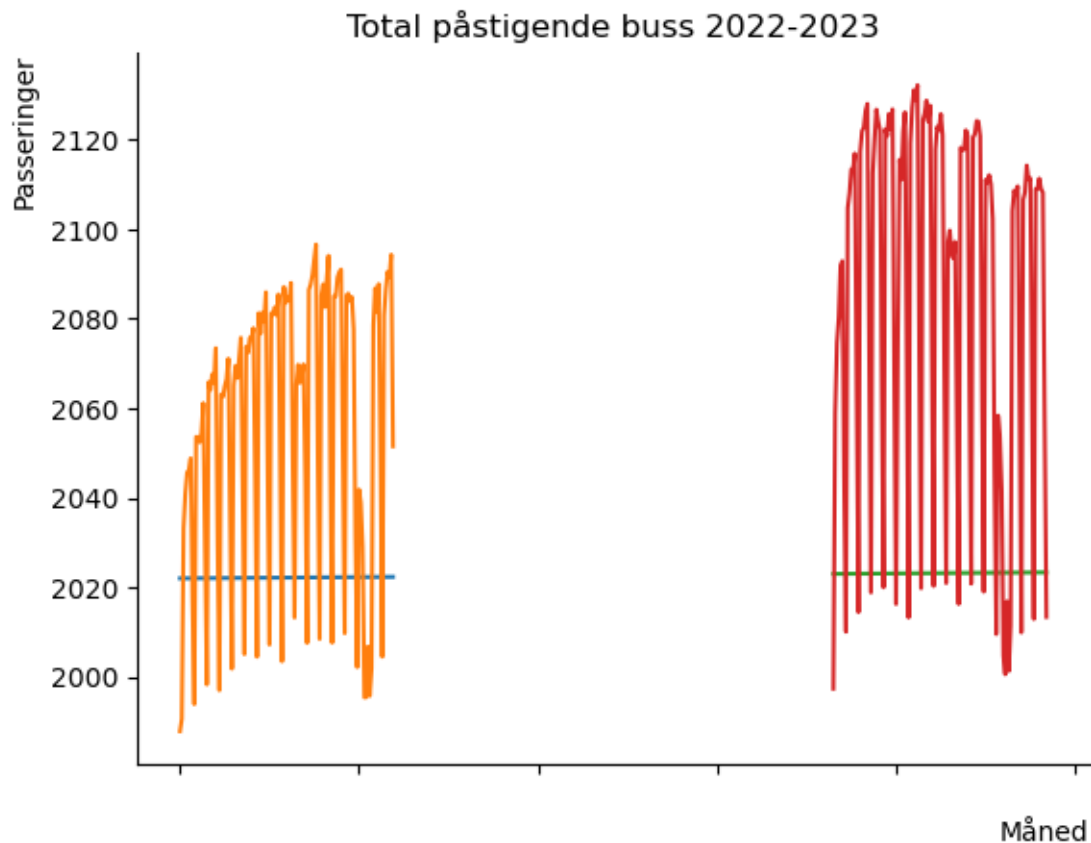
fig, ax = create_ax()

# plott funksjonen
ax.plot(df_buss22)
ax.plot(df_buss23)

# tittel
ax.set_title('Total påstigende buss 2022-2023')

Text(0.5, 1.0, 'Total påstigende buss 2022-2023')

```



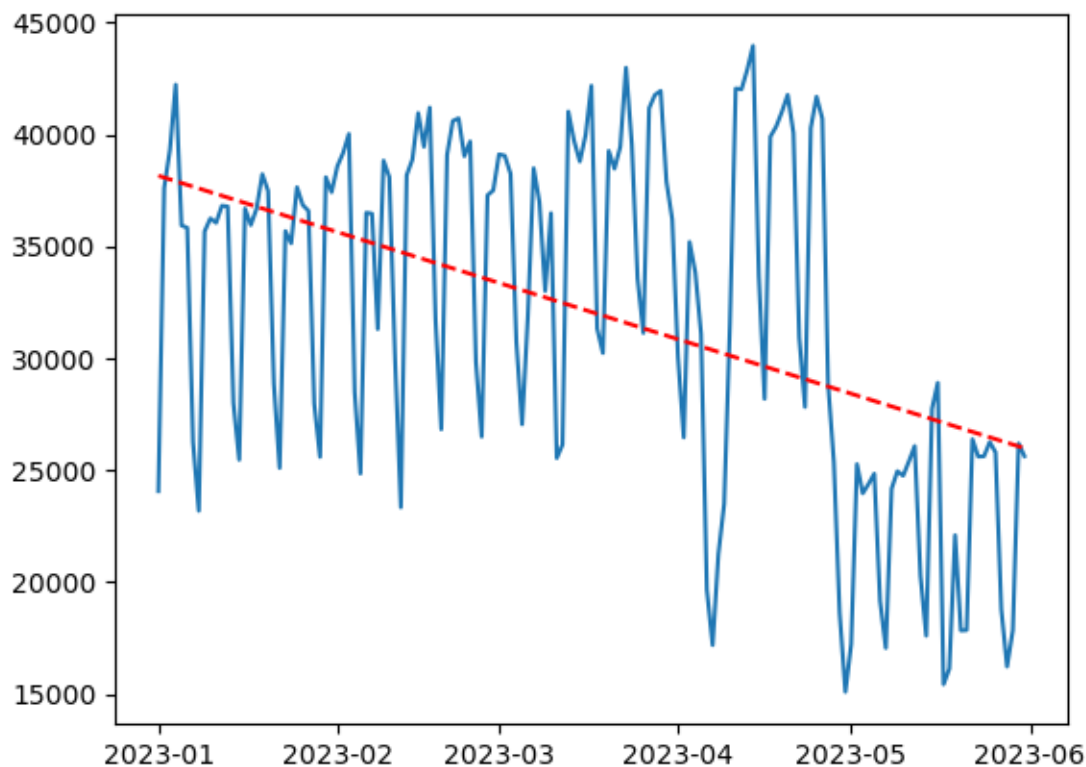
Sammneligner vi busstrafikken i 2023 med 2022 ser vi at mengden passasjerer påbussrutene på Tromsø-Øya har økt mye, som er i samsvar med nedgangen av personbiler i trafikken.

```
plt.plot(veitrafikk23['Dato'], veitrafikk23['Trafikkmengde'], label = 'trafikk')
X = pd.to_numeric(veitrafikk23['Dato']).values.reshape (-1, 1)
y = veitrafikk23['Trafikkmengde'].values.reshape(-1, 1)

regressor = LinearRegression()
regressor.fit(X, y)

reg = regressor.predict(X)

plt.plot(veitrafikk23['Dato'], reg, color = 'red', linestyle = '--', label = 'Regresjonsli
```



Denne regresjonslinjen viser trenden i biltrafikken på innpasseringsveiene til Tromsø-øya. Om vi velger å tro på trenden vil personbiltrafikken fortsette å minke, noe som tilsier at bypolitikken om å senke personbiltrafikken har virkning.

```
#data tatt fra: [^1]https://bpsnord.no/bypakke-tenk-tromso/
data={'fosilbil':[12,9.60,36,28.80], 'elbil':[12,4.80,36,14.40]}

bompenger=pd.DataFrame(data, index=['Kostnad', 'Kosnadmavtale', 'rushkostnad', 'rushkostnad'])

bompenger['fosilbil'] = bompenger['fosilbil'].astype(int)
bompenger['elbil'] = bompenger['elbil'].astype(int)
```

```
bompenger
```

| | fosilbil | elbil |
|---------------|----------|-------|
| Kostnad | 12 | 12 |
| Kosnadmavtale | 9 | 4 |

| | fosilbil | elbil |
|--------------------|----------|-------|
| rushkostnad | 36 | 36 |
| rushkostnadmavtale | 28 | 14 |

Her vises hvor mye en vanlig personbil og elbil må betale hver gang den passerer en bomstasjon. Med et tak på belastning etter 80 passeringer per måned kan vi finne ut hvor mye en maks må betale i måneden bassert på om ikke har auto-pass avtale, om du har autopass avtale men vanlig personbil, eller har elbil og auto-pass avtale. Det er også regnet ut hva en må betale om en passerer en bomstasjon om lag 2 ganger daglig.

```
bompenger['fosilbil_60'] = bompenger['fosilbil'] * 60
bompenger['fosilbil_60'] = bompenger['fosilbil_60'].astype(int)
bompenger['elbil_total_60'] = bompenger['elbil'] * 60
bompenger['elbil_total_60'] = bompenger['elbil_total_60'].astype(int)
bompenger['fosilbil_total_80'] = bompenger['fosilbil'] * 80
bompenger['fosilbil_total_80'] = bompenger['fosilbil_total_80'].astype(int)
bompenger['elbil_total_80'] = bompenger['elbil'] * 80
bompenger['elbil_total_80'] = bompenger['elbil_total_80'].astype(int)
bompenger
```

| | fosilbil | elbil | fosilbil_60 | elbil_total_60 | fosilbil_total_80 | elbil_total_80 |
|--------------------|----------|-------|-------------|----------------|-------------------|----------------|
| Kostnad | 12 | 12 | 720 | 720 | 960 | 960 |
| Kosnadmavtale | 9 | 4 | 540 | 240 | 720 | 320 |
| rushkostnad | 36 | 36 | 2160 | 2160 | 2880 | 2880 |
| rushkostnadmavtale | 28 | 14 | 1680 | 840 | 2240 | 1120 |

I denne tabllen kommer det helt klart fram hvilke alternativer som er mer lønnsomt dersom man må bruke bil ofte.

```
#Tall tatt fra: [^2]'https://finanssans.no/gjennomsnittsl%C3%B8nn-i-norge'
inntekter={'inntekter':[27920,35360,39230,42590,45990,49520,53710,59700,70070,101600]}

inntektsdesiler=pd.DataFrame(inntekter, index=['Desil1','Desil2', 'Desil3', 'Desil4', 'Desil5'])

inntektsdesiler['inntekter'] = inntektsdesiler['inntekter'].astype(int)

inntektsdesiler['fosilbil60mavtale'] = inntektsdesiler[inntekter] - 520

inntektsdesiler['elbil60mavtale'] = inntektsdesiler[inntekter] - 240
```

```

inntektsdesiler['foslbilrush'] = inntektsdesiler['inntekter'] - 1680

inntektsdesiler['elbilrush'] = inntektsdesiler['inntekter'] - 840

```

```

/tmp/ipykernel_57371/18754662.py:8: FutureWarning: Passing a dict as an indexer is deprecated
    inntektsdesiler['fosilbil60mavtale'] = inntektsdesiler[inntekter] - 520
/tmp/ipykernel_57371/18754662.py:10: FutureWarning: Passing a dict as an indexer is deprecated
    inntektsdesiler['elbil60mavtale'] = inntektsdesiler[inntekter] - 240

```

```

#[^3]tall tatt fra :https://www.ssb.no/transport-og-reiseliv/landtransport/statistikk/kjor
avg_km={'km':[11097]}

```

```

km_personbil=pd.DataFrame(avg_km)

```

```

#[^5]https://www.toi.no/getfile.php?mmfileid=74283
km_personbil['km'] = km_personbil['km'].astype(int)
km_personbil['km_per_dag'] = km_personbil['km'] /360
km_personbil['km_kostnad'] = km_personbil['km'] /12*3.03 #regner ut månedlige kostnader per
km_personbil

```

| | km | km_per_dag | km_kostnad |
|---|-------|------------|------------|
| 0 | 11097 | 30.825 | 2801.9925 |

Kostnad assosiert med å ha bil, ikke inkludert bompengavgift.

```

inntektsdesiler['personbil_total'] = inntektsdesiler['inntekter'] - 2801.99
inntektsdesiler['elbil_total'] = inntektsdesiler['elbil60mavtale'] - 2801.99
inntektsdesiler

```

| | index | inntekter | fosilbil60mavtale | elbil60mavtale | foslbilrush | elbilrush | personbil_total | elbil_to |
|---|--------|-----------|-------------------|----------------|-------------|-----------|-----------------|----------|
| 0 | Desil1 | 27920 | 27400 | 27680 | 26240 | 27080 | 25118.01 | 24878.01 |
| 1 | Desil2 | 35360 | 34840 | 35120 | 33680 | 34520 | 32558.01 | 32318.01 |
| 2 | Desil3 | 39230 | 38710 | 38990 | 37550 | 38390 | 36428.01 | 36188.01 |
| 3 | Desil4 | 42590 | 42070 | 42350 | 40910 | 41750 | 39788.01 | 39548.01 |
| 4 | Desil5 | 45990 | 45470 | 45750 | 44310 | 45150 | 43188.01 | 42948.01 |
| 5 | Desil6 | 49520 | 49000 | 49280 | 47840 | 48680 | 46718.01 | 46478.01 |

| | index | inntekter | fossilbil60mavtale | elbil60mavtale | fossilbilrush | elbilrush | personbil_total | elbil_to |
|---|---------|-----------|--------------------|----------------|---------------|-----------|-----------------|----------|
| 6 | Desil7 | 53710 | 53190 | 53470 | 52030 | 52870 | 50908.01 | 50668.01 |
| 7 | Desil8 | 59700 | 59180 | 59460 | 58020 | 58860 | 56898.01 | 56658.01 |
| 8 | Desil9 | 70070 | 69550 | 69830 | 68390 | 69230 | 67268.01 | 67028.01 |
| 9 | Desil10 | 101600 | 101080 | 101360 | 99920 | 100760 | 98798.01 | 98558.01 |

```
def create_ax():
    fig, ax = plt.subplots()
    ax.set_ylabel('Passeringer', loc='top')
    ax.set_xlabel('Måned', loc='right')
    ax.spines['top'].set_color('none')
    ax.spines['right'].set_color('none')

    return fig, ax

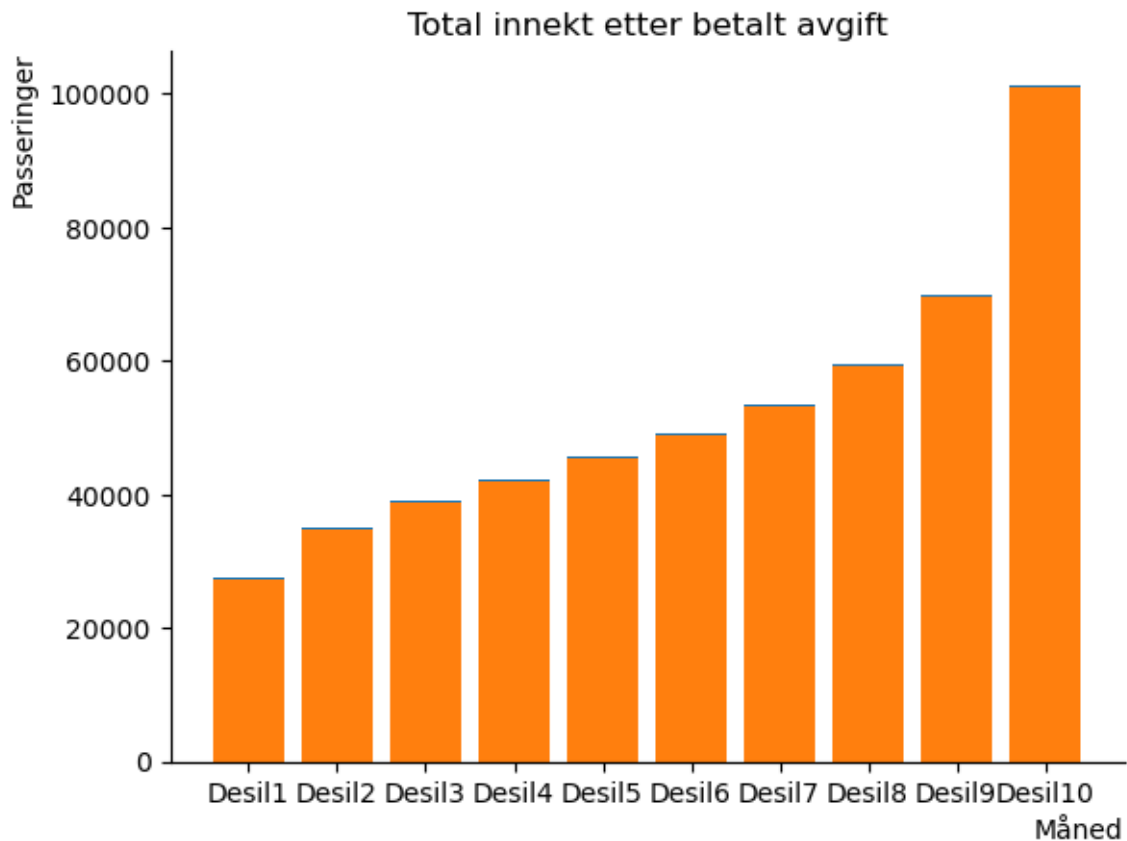
fig, ax = create_ax()

ax.bar(inntektsdesiler['index'], inntektsdesiler['elbil60mavtale'])

ax.bar(inntektsdesiler['index'], inntektsdesiler['fossilbil60mavtale'])

# tittel
ax.set_title('Total innekt etter betalt avgift')
```

```
Text(0.5, 1.0, 'Total innekt etter betalt avgift')
```

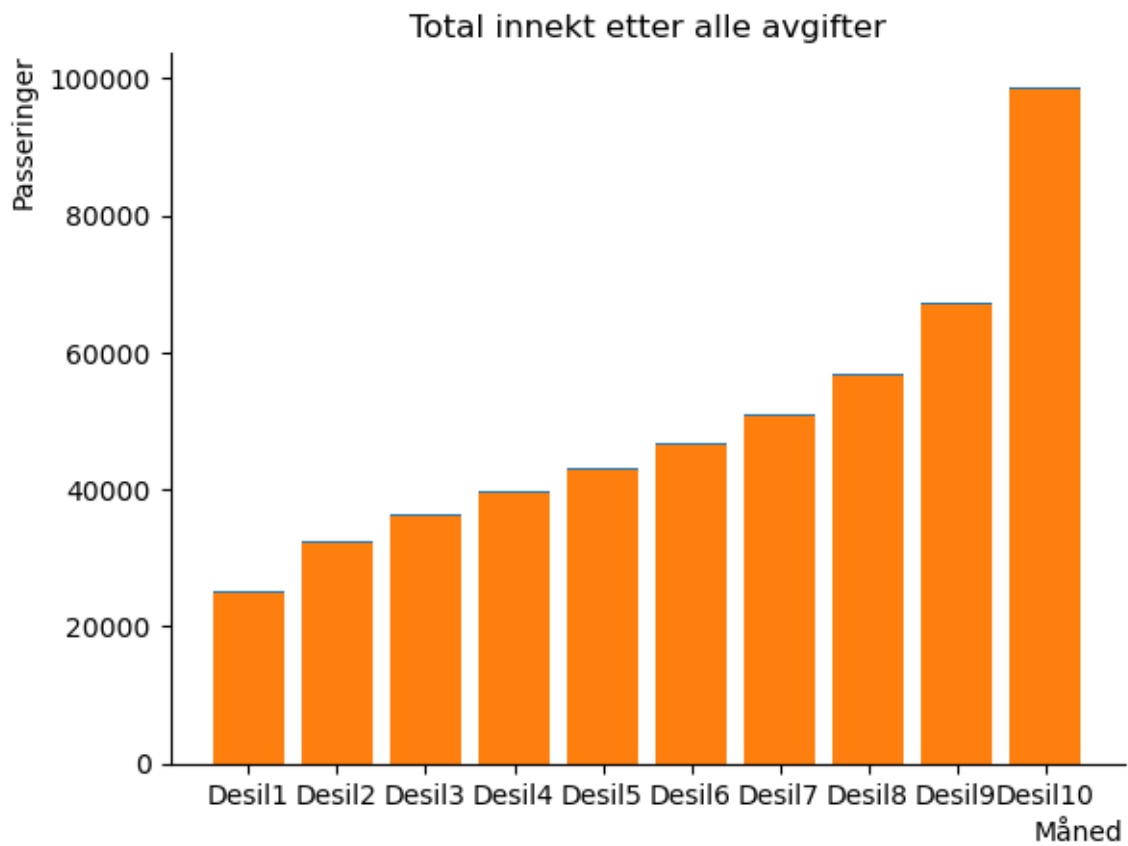


```
def create_ax():  
    fig, ax = plt.subplots()  
    ax.set_ylabel('Passeringer', loc='top')  
    ax.set_xlabel('Måned', loc='right')  
    ax.spines['top'].set_color('none')  
    ax.spines['right'].set_color('none')  
  
    return fig, ax  
  
fig, ax = create_ax()  
  
ax.bar(inntektsdesiler['index'], inntektsdesiler['personbil_total'])  
  
ax.bar(inntektsdesiler['index'], inntektsdesiler['elbil_total'])  
  
# tittel
```

```
ax.set_title('Total innekt etter alle avgifter')

print(inntektsdesiler[['inntekter', 'personbil_total', 'elbil_total']])
```

| | inntekter | personbil_total | elbil_total |
|---|-----------|-----------------|-------------|
| 0 | 27920 | 25118.01 | 24878.01 |
| 1 | 35360 | 32558.01 | 32318.01 |
| 2 | 39230 | 36428.01 | 36188.01 |
| 3 | 42590 | 39788.01 | 39548.01 |
| 4 | 45990 | 43188.01 | 42948.01 |
| 5 | 49520 | 46718.01 | 46478.01 |
| 6 | 53710 | 50908.01 | 50668.01 |
| 7 | 59700 | 56898.01 | 56658.01 |
| 8 | 70070 | 67268.01 | 67028.01 |
| 9 | 101600 | 98798.01 | 98558.01 |



En månedlig kostnad på 240 og 520 kan for seg selv virke ganske lite, men det er viktig å tenkte på at for en normal norsk husstand vil dette lang i fra være den eneste kostnaden, skatt er selvfølgelig en stor bulk i månedsinntekten til de fleste husstander, og de fleste har lån på hus og andre lån som nedbetales månedlig, kostnad på bil som vi har regnet ut er i tillegg 2081 kr månedlig, og de koster også mye å kjøpe. Om vi legger sammen alle disse månedlige kostnader ser vi mer og mer hvilken påvirkningskraft denne ekstra kostnaden vil ha, særlig for de desilene med lavere inntekt. For desil 1 er denne månedlige avgiften 10,90% av den månedlige inntekten, dette vil bare minke for de desilene som tjener mer.

En vanlig månedsbillet på buss for en voksen person koster månedlig 580 kroner, dette er mer en man betaler månedlig med en auto-pass avtale, men det finnes ingen flere kostnader en den månedlige kostnaden.

Som en del av bybilde politikken får elbiler mindre avgift dersom de også har auto-pass avtale, men denne rabatten gjelder kun for de som har råd til å oppgradere til en elbil. det er derfor de lavere inntektsdesilene som i utgangspunktet hadde mest bruk for denne rabatten som mest sannsynlig ikke får ta denne rabatten i bruk. Kunne en annen avgiftsløsning vært nyttig her?

I Finnland blir man bøtelagt prosentvis etter sin inntekt, om de øvrige desilene med mer inntekt hadde fått en høyere avgift basert på sin inntekt ville dette vært med på å jevnere fordele denne samfunnsbyrden da de som har minst inntekt ville betalt den samme prosenten av sin inntekt men i praksis hadde betalt mindre. Det finnes nok mange utfordringer med denne løsningen, men det ville vært enda større intensiv for å ta i bruk kollektivtrafikk i storbyene da denne løsningen fremdeles bare ville ha vært i bruk i trafikken innad storbyene.

Kapittel 3 - Bilisters tilpassning til bompenger.

For å finne hvilke tall vi skal sette inn i figuren kan vi se på tall vi regnet ut tidligere. Vi vet prisen for elbil per måned, og vi vet prisen for fossilbil per måned. Om vi regner ut gjennomsnittet av disse, deler gjennomsnittet på antal dager i året, deler det på gjennomsnittet km kjørt hver dag kommer vi fram til en kostnad etter at bompenger er innført. $240+520/2 = 380/31/30,825 = 0,397$ (pris bompenger per km.)

[⁵]Vi legger så det til prisen per km uten bompenger og får en ny pris per km inkludert bompenger. $0,397 + 3,03 = 3,427 = \text{pris per km}$

[²]Medianlønnen finner vi om vi legger sammen desil 5 og 6 og deler på 2. da får vi $45990+49520/2 = 47755$. $47755*/31$ gir oss et daglig busjett på 1540 når vi runder ned.

```
def budsjett(M, p1, p2, x1):  
    return M/p2 - p1*x1/p2  
  
x = np.linspace(0.001,1000,1000)
```

```

def create_ax():
    fig, ax = plt.subplots()
    ax.set_ylabel('$x_2$', loc='top', rotation=0)
    ax.set_xlabel('$x_1$', loc='right')
    ax.set_xlim(0,1000)
    ax.set_ylim(0,2000)
    ax.spines['top'].set_color('none')
    ax.spines['right'].set_color('none')

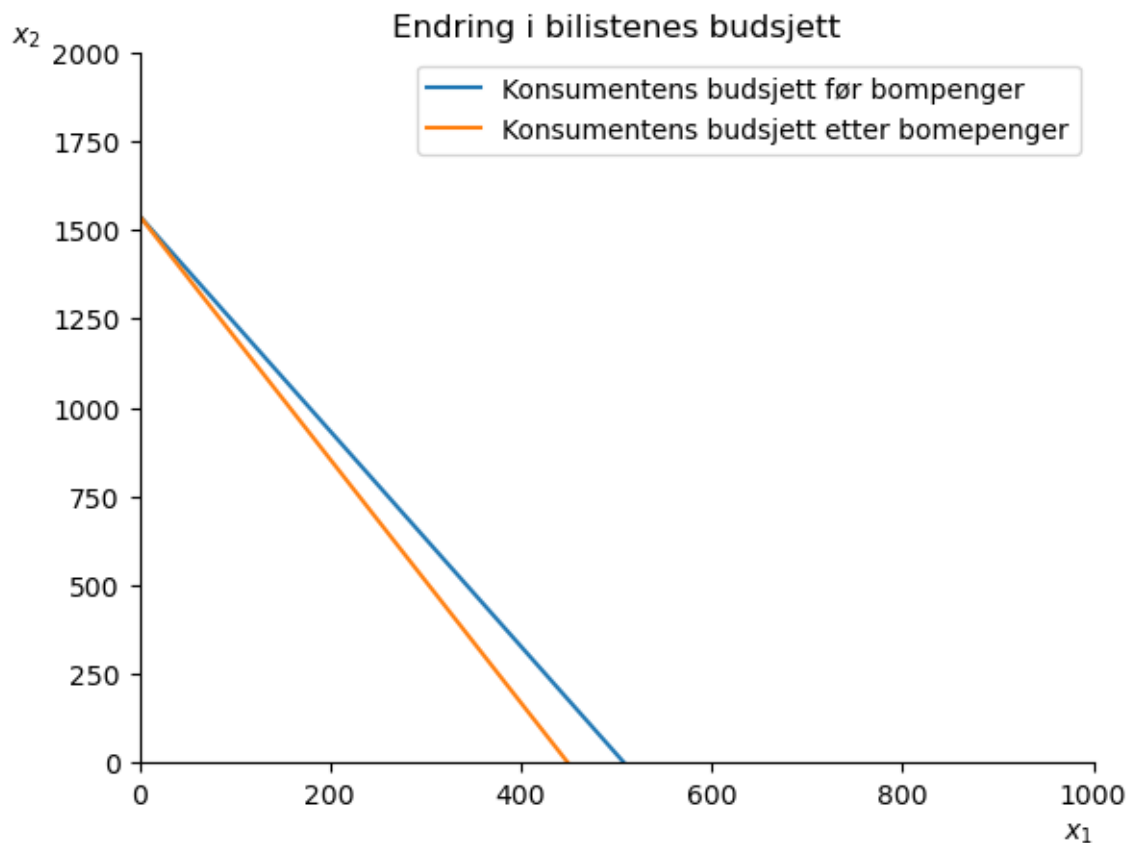
    return fig, ax

fig, ax = create_ax()

ax.plot(x, budsjett(1540, 3.03, 1, x), label = 'Konsumentens budsjett før bompenger')
ax.plot(x, budsjett(1540, 3.427, 1, x), label = 'Konsumentens budsjett etter bompenger')

ax.set_title('Endring i bilistenes budsjett')
ax.legend(loc = 'best')
txt = ''
fig.text(.5, .01, txt, ha='center');

```



```

bta_1, bta_2, gma_1, gma_2, U_0 = sp.symbols('\u03B2_1 \u03B2_2 \gamma_1 \gamma_2 U_0', real=True, positive=True)
x_1, x_2, p_1, p_2, m = sp.symbols("x_1 x_2 p_1 p_2 m", real=True, positive=True)

nytte_sg = bta_1*sp.log(x_1-gma_1) + (1-bta_1)*sp.log(x_2-gma_2)

x_1_eq = gma_1 + (bta_1/p_1)*(m-p_1*gma_1-p_2*gma_2)
x_2_eq = gma_2 + (bta_2/p_2)*(m-p_1*gma_1-p_2*gma_2)

V = nytte_sg.subs([(x_1, x_1_eq), (x_2, x_2_eq)])

ind_k = sp.solve(nytte_sg-U_0, x_2)[0]

def ind_k_fn(bta_1,gma_1,gma_2,x_1,U_0):
    return gma_2 + np.exp((bta_1*np.log(x_1-gma_1)-U_0)/(bta_1-1))

x_1_eq_num = float(x_1_eq.subs([(bta_1, 0.5), (gma_1, 1), (gma_2, 1), (m, 1540), (p_1, 3.0)]))

```



```

x_2_eq_num = float(x_2_eq.subs([(bta_2, 0.5), (gma_1, 1), (gma_2, 1), (m, 1540), (p_1, 3.0
nytte_sg_num = float(V.subs([(bta_1, 0.5), (bta_2, 0.5), (gma_1, 1), (gma_2, 1), (m, 1540),

x_1_eq_ny = float(x_1_eq.subs([(bta_1, 0.5), (gma_1, 1), (gma_2, 1), (m, 1540), (p_1, 3.42
x_2_eq_ny = float(x_2_eq.subs([(bta_2, 0.5), (gma_1, 1), (gma_2, 1), (m, 1540), (p_1, 3.42
nytte_sg_ny = float(V.subs([(bta_1, 0.5), (bta_2, 0.5), (gma_1, 1), (gma_2, 1), (m, 1540),

```

[⁴] Ved bruk av et fast daglig budsjett blir Stone Geary det naturlige valget da vi alokrerer alle tilgjengelige resurser i de godene vi behøver. Fordelen kommer i form av at vi kan andre på preferanser, om vi vil ha 1 av hver gode, 1 av en gode og halv av en annen vil man kunne justere dette og derfor bestemme hvilken resurs man verdsetter mer.

```

X = np.linspace (1.01,1200,2000)

fig2, ax = create_ax()
ax.set(xlim=(0,1200))
ax.set(ylim=(0,1200))
ax.set_ylabel('Andre goder', loc='top', rotation=0)
ax.set_xlabel('Antall km kjørt', loc='right')

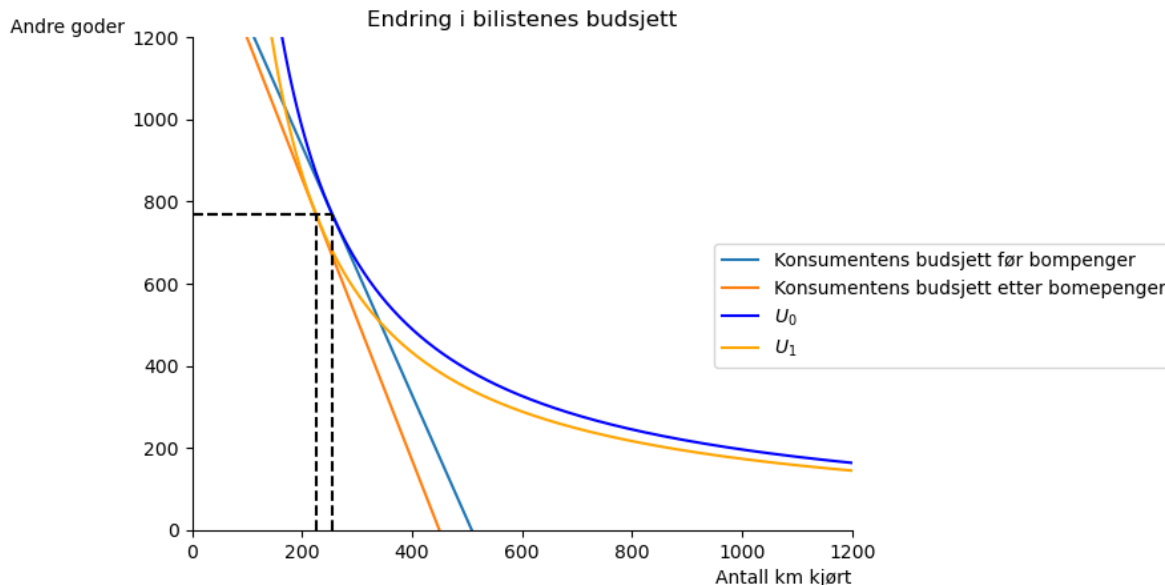
ax.plot(x, budsjett(1540, 3.03, 1, x), label = 'Konsumentens budsjett før bompenger')
ax.plot(x, budsjett(1540, 3.427, 1, x), label = 'Konsumentens budsjett etter bompenger')
ax.plot(X, ind_k_fn(.5,1,1,X,nytte_sg_num), color='blue',label='$U_0$')
ax.plot(X, ind_k_fn(.5,1,1,X,nytte_sg_ny), color='orange', label='$U_1$')

ax.vlines(x_1_eq_num, 0, x_2_eq_num, ls='dashed', color='black')
ax.hlines(x_2_eq_num, 0, x_1_eq_num, ls='dashed', color='black')
ax.vlines(x_1_eq_ny, 0, x_2_eq_ny, ls='dashed', color='black')
ax.hlines(x_2_eq_ny, 0, x_1_eq_ny, ls='dashed', color='black')

ax.set_title('Endring i bilistenes budsjett')

ax.legend(bbox_to_anchor=(1.5,0.6));

```



Observasjonen som kan gjøres fra visualiseringen av kostnader før og etter bompengeneinnføringen er at det etter innføringen koster mer å kjøre bil. Vi kan se dette basert på at vi ikke har fått mer råd til andre goder.

Basert på det budsjettet vi regnet oss fram til viser grafen oss hvilken mulighet vi har for å maksimere budsjettet vårt med den nye prisen. For å videre endre på antall kilometer kjørt er vi nødt til å trekke fra andre goder, å for å få mer av andre goder må man trekke fra antall km kjørt.

Kapittel 4 - Konklusjon.

Basert på funnene gjort i denne analysen kan vi sterkt fastsette at innføring av bompenger som en ekstra kostnad vil påvirke både økonomien og oppførselen til bilistene. Dagen den ble innført gikk trafikken i de berørte områdene ned en god prosentandel, og siden vi senere observerte en minking i antall biler på samme veistrekninger ser vi basert på økningen i antall busspassasjerer at flere biler trolig har gjort en vaneendring.

Innføringen av bompenger påvirker forskjellige innteksgrupper forskjellig, det som fører til vaneendringer hos de mest lavtlønnede i Norge vil ikke nødvendigvis føre til samme endring hos de mest høylønnede da den totale prisen prosentmessig ikke blir merket på samme måte. De mer lavtlønnede har heller ikke lik mulighet til å tilpasse seg betingelsene for å kunne ta imot de største rabattene på bompenger da inngangsprisen på dette ofte kan være høye, blant annet å skaffe seg en elbil, som enda er ganske nytt ikke er direkte billig.

Økingen i kostnader assosiert med å eie bil å kjøre i storbyer har ført til et strammere budsjett og de fleste bilister må tilpasse seg da det per kilometer koster mer å kjøre. Dette vil i det lange løp føre til at flere personer i storbyene velger kollektivtrafikk framfor å kjøre selv, og derfor vil trafikken minke som i tur gjør kollektivtrafikken mer effektiv.

Kilder:

[¹]<https://bpsnord.no/bypakke-tenk-tromso/>

[²]<https://finanssans.no/gjennomsnittsl%C3%B8nn-i-norge>

[³]<https://www.ssb.no/transport-og-reiseliv/landtransport/statistikk/kjorelengder>

[⁴]https://www.monash.edu/__data/assets/pdf_file/0004/925456/estimating_residential_water_demand_using_geary_functional_form_the_case_of_sri_lanka.pdf

[⁵]<https://www.toi.no/getfile.php?mmfileid=74283>