

Metodbeskrivning - Läget i kommunen

Metodbeskrivning av den Kommununika Logikstyrda rapporten med Ai-Stöd.

Innehåll

Rapportens grundläggande byggstenar.....	3
Excelformuläret – 'Beställningsformuläret'	3
All_metadata.xlsx	4
Rapportfunktioner.R	5
"Läget – Get data.Rmd"	5
Översikt.....	5
Val av år och nyckeltal	6
Kommunlista	6
Anpassar anropen till Koladas API	6
Hämtar data från Kolada API	6
Fokusering på totalvärden och komplettering av saknade kombinationer	7
Separering av riket och "Alla kommuner (ovägt medel)"	7
Beräkning av kvartiler, percentiler och deciler.....	7
Förändring över tid	8
Inkoppling av metadata	8
Identifiering av "fish" (ihållande låg nivå) och "bird" (ihållande hög nivå).....	8
Rangordning av förbättringar och försämringar	9
Urval av kommuner som lyckats bra per nyckeltal.....	10
Jämförelsegrupper efter befolkningsstorlek.....	10
Slutliga utdatafiler.....	11
Läget – Generera rapport.Rmd	12
Översikt.....	12
Rapportformat och layout.....	12
Val av kommun och grundläggande inställningar	13
Inläsning av paket och externa funktioner	13
Inläsning av färdiga datafiler	13
Titel, datum, förord och signaturer.....	14

AI-genererad sammanfattning (exekutiv sammanfattning)	14
Avsnittet “Det här innehåller rapporten”	15
Huvudloopen: ett avsnitt per område.....	15
Fokusnyckeltal inom varje område	17
Diagram och förklarande texter för fokusnyckeltal	18
Avsnittet “Om rapporten”.....	19
Läget – Generate reports.R	19

Rapportens grundläggande byggstenar

Rapporten består av en samling filer som är beroende av varandra för att rapporterna ska kunna skapas. Nedan listas samtliga filer och deras syfte i detalj:

Excelformuläret – 'Beställningsformuläret'

I den här versionen av rapporten görs alla huvudsakliga inställningar i ett excelformulär. Tanken är att även en kod-ovan användare ska kunna skapa grunden till en ny rapport med ett eget urval av nyckeltal eller ämnesområde.

Excelformuläret har i huvudsak fem flikar:

- **År** – här behöver 'Fokusår' och 'Antal år utöver fokusåret' anges.
 - Fokusår: Sista tillgängliga året av alla nyckeltal.
 - Antal år utöver fokusåret: Antal år före fokusåret till T-2 där T avser det sista året för de/det nyckeltal med äldsta sista år i tidsserien.
 - Exempel:
 - Rapporten innehåller 20 nyckeltal.
 - 5 av dem har sista tillgängliga år 2023.
 - 10 har sista tillgängliga år 2024.
 - 5 har sista tillgängliga år 2025.
 - Fokusåret är då **2025**.
 - 'Antal år utöver fokusåret' blir **4** eftersom $2023 - 2 = 2021$. Åren utöver fokusåret är då 2021, 2022, 2023, 2024. Fyra (4) stycken. Observera att detta avser minsta antal. Vill man visa en längre tidsserie går det bra.
 - Är det lång tid mellan äldsta och yngsta senaste år bland nyckeltalen bör man överväga om man vill redovisa dem i samma rapport.
- **Titel och förord** – här anges rapportens titel, datum då den blir publicerad och plats för upp till sex stycken. Förordet bör vara mellan 200 till 400 ord beroende på hur många signaturer som anges.
- **Signaturer** – Plats för fyra signaturer med för och efternamn samt titel.
- **Kommuner** – En lista över samtliga 290 kommuner. Markera med en etta (1) för de kommuner som du önskar generera en rapport för.

- **Nyckeltal** – Här anges vilka nyckeltal som ska inkluderas i rapporten. Den här fliken innehåller i sin tur åtta kolumner där metadata för varje enskilt nyckeltal fylls i. Metadata i den här fliken är av sådan karaktär att den varken finns i Koladas API eller i adminverktyget för Kolada. Informationen på denna flik bidrar mer till rapportens struktur. Om rapporten syftar till en viss sida i Koladaverktyget Jämföraren är det rimligt att efterlikna den strukturen men det är inget tvång. Kolumnerna fylls i enligt:
 - **ntid:** nyckeltalets id till exempel 'N11800'
 - **kortnamn:** kortnamn på nyckeltalet. Detta är mest till för att se vilka nyckeltal man lagt in. I rapporten används de namnet som nyckeltalet har i apiet.
 - **rubrik:** rubriken avgör nyckeltalets områdestillhörighet. Antalet uika rubriker bestämmer hur många områden som redovisas i rapporten. Varje område får en tabell och möjlighet att hitta fokusnyckeltal.
 - **rubrikordning:** i vilken ordning ska de olika rubrikerna (områdena) redovisas? Observera att varje unikt rubriknamn måste ha samma siffra i rubrikordningen.
 - **underrubrik:** Här finns det möjlighet att lägga in underrubriker i tabellerna. Du måste inte ha underrubrik. Om du för en enskild rubrik (område) inte vill ha underrubriker lämnar du bara den här kolumnen tom för dessa nyckeltal.
 - **underrubrikordning:** Fyller samma funktion som rubrikordning men avser ordningen på underrubrikerna under varje enskild rubrik.
 - **order:** i vilken ordning ska nyckeltalen listas i tabellen?
 - Exempel: säg att ett nyckeltal har rubrikordning 2 underrubrikordning 3 och order 2 så kommer detta nyckeltal kommer visas i rapportens andra tabell, under den tredje underrubriken ovanifrån och det andra nyckeltalet under underrubriken.
 - **viktad:** Ska nyckeltalet kunna visas utanför tabellen? Har nyckeltalet lov att omnämnas i huvudsammanfattningen och rubriksammanfattningen och väljas ut att vara ett fokusnyckeltal. Eller kommer det bara visas inuti tabellen?

All_metadata.xlsx

Ett utdrag av samtliga nyckeltal i Koladas metadata – oavsett om de ingår i rapporten eller ej. Filen används till att hämta information om antalet decimaler för nyckeltalen samt 'lowgood' information. Alltså huruvida låga eller höga värden är att föredra för det enskilda nyckeltalet.

Rapportfunktioner.R

Ett externt R-script där större funktioner lagras. Här finns funktioner som

- Skapar linjediagrammen för fokusnyckeltal
- Anger färg på fokusnyckeltalens ram
- Regelstyr förklaringen till varför ett nyckeltal valts till fokusnyckeltal
- Sköter sammanfattning av data samt kommunicerar med AI-leverantörens API. I nuläget är valet av språkmodell/ai-leverantör till viss del hårdkodat och kan behöva handpåläggning i framtida rapporter.

Förutom filerna nämnda ovan finns också en separat .txt-fil med API-nyckeln som krävs för anrop till språkmodellernas API:er.

”Läget – Get data.Rmd”

Översikt

Det här delscriptet används för att hämta in, komplettera och bearbeta data som ligger till grund för rapporten. Scriptet gör ingen layout eller rapportproduktion i sig, utan tar fram den datafil som nästa script använder för att bygga själva rapporterna.

Scriptets funktion kan delas upp i följande huvudsteg:

1. Läs in inställningar och metadata (vilka år, vilka nyckeltal, beskrivningar m.m.).
2. Hämta statistik från Koladas API för alla kommuner, alla valda nyckeltal och alla aktuella år.
3. Komplettera data så att det finns rader även där värden saknas, och beräkna förändringar över tid.
4. Beräkna jämförelsemått (kvartiler, percentiler, deciler, rankningar).
5. Identifiera särskilt intressanta nyckeltal för de enskilda kommunerna (t.ex. nyckeltal med stora förbättringar/försämringar och nyckeltal vars värde legat stilla en längre tid på en fördelaktig eller ofördelaktig nivå).
6. Skapa jämförelsegrupper efter befolkningsstorlek.
7. Spara färdiga underlagsfiler som används i rapportscriptet.

Val av år och nyckeltal

Först läser scriptet in en inställningsfil i Excel (Definitioner.xlsx). Där anges fokusår (t.ex. 2024) och hur många år bakåt som ska ingå. Utifrån detta byggs en årslista, t.ex. 2019 - 2024, som används i alla senare steg.

Scriptet läser också in två metadatafiler:

- En manuellt skapad excelfil med de nyckeltal som ska ingå i rapporten (Definitioner.xlsx, fliken Nyckeltal). Den här filen styr exakt vilka nyckeltal som hämtas.
- En excelfil (all_metadata.xlsx) som är exporterad från Koladas system. Den innehåller ytterligare metadata, t.ex. beskrivningar, information om vad som är "bra"/"dåligt" (Lowgood) och antal decimaler.

Viss överflödig information rensas bort, och dubletter tas bort så att varje nyckeltal bara förekommer en gång.

Kommunlista

Kommunlistan hämtas direkt från Koladas API, där alla kommuner av typen "K" tas ut. Därefter lägger scriptet till två extra "kommuner":

- Alla kommuner (ovägt medel)
- Riket

Dessa används senare i rapporterna och sparas separat så att de inte stör jämförelserna mellan enskilda kommuner.

Anpassar anropen till Koladas API

För att inte riskera att göra för stora anrop delas nyckeltalslistan upp med nyckeltal upp i grupper om 15 nyckeltal. För varje sådan grupp hämtas data kommun för kommun, för alla år i årslistan.

Hämtar data från Kolada API

Själva hämtningen sker i två nästlade loopar:

1. Ytterst: för varje delmängd av nyckeltal (15 åt gången).
2. Innerst: för varje kommun (inklusive "Riket" och "Alla kommuner (ovägt medel)").

För varje kombination av:

- kommun,
- batch av nyckeltal och
- alla år i årslistan

bygger scriptet **upp en API-adress och** skickar en förfrågan till Kolada. Om svaret är okej:

- Plockas alla värden ut per:
 - nyckeltal,
 - kommun,
 - år,
 - kön (T = totalt, K = kvinnor, M = män),
 - och själva värdet.

Varje värde blir en rad som läggs till i en växande tabell. Resultatet efter detta steg är en stor datafil som innehåller alla hämtade värden uppdelat på kommun, nyckeltal, år och kön.

Fokusering på totalvärden och komplettering av saknade kombinationer

Rapporten arbetar i den här versionen inte med könsuppdelad statistik. Därför filtreras data ned till totalvärden (gender = "T").

När Koladas API inte har någon post alls för en viss kombination av kommun–nyckeltal–år saknas raden helt. För att kunna göra fullständiga jämförelser skapas först en “utopi-tabell” med alla tänkbara kombinationer av nyckeltal, kommuner och år i årslistan.

Den verkliga data från hämtningen matchas mot denna fullständiga tabell. Kombinationer där inget värde finns får värdet NA, på samma sätt som registrerade bortfall/sekretess.

På så sätt säkerställs att det finns en rad för varje kommun–nyckeltal–år, oavsett om värdet finns eller inte.

Separering av riket och “Alla kommuner (ovägt medel)”

Innan vidare beräkningar genomförs plockas Riket och “Alla kommuner (ovägt medel)” ut i en separat tabell. De tas bort från den huvudsakliga data som används för att beräkna kvartiler, percentiler, deciler och liknande, eftersom de annars skulle påverka jämförelsen mellan enskilda kommuner.

Beräkning av kvartiler, percentiler och deciler

För varje nyckeltal och år beräknas nyckeltalsvärdets kvartil, percentil och decil.

Detta används dels i rapporterna (t.ex. för att kunna färgkoda tabeller), dels i senare analyser, t.ex. för att hitta nyckeltal där en kommun legat mycket lågt eller högt jämfört med andra under lång tid.

Förändring över tid

För varje kombination av kommun och nyckeltal beräknas förändringen från år till år:

- `change_since_last` = skillnaden mellan aktuellt år och föregående år.
- `change_second_last` = skillnaden mellan aktuellt år och året två år bakåt.

På så sätt kan vi hantera situationer där det saknas värde ett enstaka år, och ändå göra en jämförelse bakåt vid behov. Förändringstalen avrundas till ett rimligt antal decimaler enligt metadata (antal decimaler som nyckeltalet normalt visas med).

Inkoppling av metadata

Därefter kompletteras data med all nödvändiga metadata:

- Från den manuella metadatafilen:
 - rubriker, underrubriker och ordning,
 - markering om nyckeltalet är viktat (används senare för att välja ut särskilt viktiga nyckeltal).
- Från exporten `all_metadata.xlsx`:
 - Nyckeltalsbeskrivning (används i nuläget inte)
 - information om vad som är “bra” respektive “dåligt” (Lowgood),
 - antal decimaler som ska användas.

Detta gör att samma datafil kan användas direkt i rapportscriptet, utan att ytterligare metadata behöver läsas in där.

Identifiering av “fish” (ihållande låg nivå) och “bird” (ihållande hög nivå)

Scriptet skapar två indikatorer per kommun och nyckeltal med arbetsnamnen “fish” och “bird”.

- “fish” – kommuner som under större delen av den studerade perioden:
 - legat i den *minst fördelaktiga* decilen under 75 % av tiden rapporten avser, givet hur nyckeltalet tolkas (Lowgood), och
 - även under senaste tillgängliga år ligger kvar i den minst fördelaktiga decilen.
- “bird” – motsvarande men för den *mest fördelaktiga* decilen.

För att få etiketten “fish” eller “bird” krävs:

- tillräckligt många år med giltiga värden,

- att minst cirka 75 % av åren ligger i “dålig” respektive “bra” decil,
- att det sista tillgängliga året följer samma mönster,
- och att nyckeltalet är markerat som viktat (d.v.s. ett centralt nyckeltal).

Denna logik används för att hitta nyckeltal där kommunen varken rör sig i fördelaktig eller ofördelaktig riktning, utan ligger konsekvent väldigt fördelaktigt eller väldigt ofördelaktigt till jämfört med andra kommuner.

Rangordning av förbättringar och försämringar

För att kunna välja ut vilka nyckeltal som ska kommenteras i rapporten beräknas hur kommunerna rör sig i förhållande till andra kommuner, inte bara i sina egna värden.

1. För varje nyckeltal och år räknas en normaliserad rank:
 - Bästa kommunen får talet 0,
 - sämsta får talet 100, med hänsyn till om “högt är bra” eller “lågt är bra” (Lowgood).
2. För varje kommun–nyckeltal jämförs den normaliserade ranken:
 - i första hand mot föregående år,
 - i andra hand mot året två år bakåt, om värde saknas året däremellan.
3. Förändringen i rank (rank_change) mäts:
 - negativ förändring → kommunen har klättrat i rankingen (gått bättre),
 - positiv förändring → kommunen har tappat i rankingen (gått sämre).
4. Samtidigt kontrolleras att värdeförändringen går i “rätt riktning”:
 - om “högt är bra” måste värdet ha ökat för att tolkas som förbättring,
 - om “lågt är bra” måste värdet ha minskat, osv.

Endast förändringar där både rank och värde rör sig åt samma håll räknas som legitima förbättringar respektive legitima försämringar. Anledningen till detta är till exempel om ett nyckeltal som fått ett bättre relativt rank-tal jämfört med andra kommuner (på grund av att det gått ännu sämre än för den valda kommunen) inte ska kunna väljas ut trots att det rent värdemässigt gått sämre. För varje kommun rangordnas dessa legitima förbättringar respektive försämringar efter storleken på förändringen, och de fem största förbättringarna får en rang 1–5 (*best_change_rank*) och de fem största försämringarna får en rang 1–5 (*worst_change_rank*). Det är dessa som senare används för att styra vilka nyckeltal rapportens AI-kommentarer fokuserar på. På så sätt undviker vi att kommentera små, marginella förändringar.

Urval av kommuner som lyckats bra per nyckeltal

Scriptet tar också fram en separat lista med kommuner som lyckats särskilt bra per nyckeltal. För att en kommun ska listas som särskilt bra för ett specifikt nyckeltal måste kommunen ha:

- ha tillräckligt många år med data (minst cirka 75 % av den studerade perioden),
- att nyckeltalet har en tydlig önskad riktning (Lowgood får inte vara lika med 2),
- att utvecklingen går i fördelaktig riktning (förbättrade värden över tid),
- att kommunen ligger i bästa kvartilen vid det senaste tillgängliga året,
- att de senaste åren inte är alltför gamla.

För varje nyckeltal väljs de kommuner ut som uppfyller kriterierna, sorterat efter hur stor den fördelaktiga förändringen varit och deras namn sparas i en separat fil. Denna information användas för att lyfta fram "föredömliga kommuner" i rapporten i de fall en kommun har fokusnyckeltal av ofördelaktig karaktär.

Jämförelsegrupper efter befolkningsstorlek

För att undvika att små kommuner jämförs direkt med storstäder skapas jämförelsegrupper baserat på befolkningsstorlek. Scriptet hämtar befolkningsdata (totalbefolkning) för fokusåret minus 1 (t.ex. 2023 om fokusår är 2024). Anledningen till detta är att scriptet kan köras även innan den allra senaste befolkningsstatistiken publicerats (skillnaden bör vara marginell i kontexten uppgifterna sedan används i).

För varje kommun beräknas skillnaden i befolkningsstorlek mot alla andra kommuner, därefter sorteras kommunerna efter hur nära de ligger i befolkningsstorlek. De 75 närmaste kommunerna i befolkning väljs ut som en jämförelsegrupp.

Resultatet är en fil där varje kommun har en lista med andra kommuner som liknar den i befolkningsstorlek. Denna används senare i rapporterna när föredömliga kommuner ska lyftas fram inom en "rimlig" storleksklass. Ju längre ut på kanterna i listan av totalbefolkning en kommun befinner sig, desto mer kommer de förmodligen skilja sig mot sina 75 "närmsta" grannar. Detta är extra tydligt i fallen Stockholm, Göteborg och Malmö.

Slutliga utdatafiler

I slutet skapas och sparas flera färdiga datafiler i RDS-format:

1. **rapport_data.rds**

Den huvudsakliga datafilen som används i rapportscriptet. Den innehåller bland annat:

- kommunnamn och kommun-ID,
- nyckeltals-ID,
- år och värde (avrundat enligt metadata),
- kvartil, decil, percentil,
- förändringar över tid (ett år och två år),
- rankning av bästa respektive sämsta förändringar (1–5),
- indikatorerna “fish” och “bird”,
- senaste år med tillgängligt värde,
- metadata som rubriker, ordning, kortnamn, beskrivning, antal decimaler, Lowgood och om nyckeltalet är viktat.

2. **rapport_data_riket_medel.rds**

En motsvarande fil för:

- Riket och “Alla kommuner (ovägt medel)”. Dessa används för att redovisa nationella jämförelsevärden.

3. **top_muni_per_ntlD.rds**

En fil som för varje nyckeltal listar de kommuner som uppfyller kriterierna för att ha förbättrats på ett fördelaktigt sätt.

4. **comparegroups.rds**

En fil som innehåller jämförelsegrupperna baserade på befolkningsstorlek för varje kommun.

Läget – Generera rapport.Rmd

Översikt

Det här scriptet används för att generera själva rapportdokumentet för en kommun, baserat på den färdiga datafil som skapats i föregående steg.

Det kan användas på två sätt:

1. En enstaka rapport
Längst upp i scriptet finns rader där man kan ange en vald kommun (t.ex. `selected_municipality <- "Jönköping"`).
När dessa rader är aktiva genereras en rapport enbart för den kommunen.
2. Många rapporter i följd
När man vill skapa rapporter för flera kommuner används i stället ett separat "loopscript" som:
 - går igenom kommunlista,
 - sätter `selected_municipality` för varje kommun,
 - och kör det här rapportscriptet en gång per kommun.

Rapportformat och layout

Rapporten produceras som ett HTML-dokument som är anpassat för att skrivas ut som PDF, med hjälp av paketet *pagedown*. I början av dokumentet anges bland annat

- att utskriftsmotorn ska använda CairoPNG (för snygga diagram och text),
- att dokumentet ska vara självförsörjande (allt inbäddat),
- att det inte ska finnas någon innehållsförteckning.

Därefter följer en längre CSS-del (stilmall) som styr utseendet:

- Banner högst upp med kommunnamn, rapporttitel och datum.
- Typsnitt, färger och marginaler för rubriker, brödtext, tabeller och boxar.
- En särskild "rounded box"-stil för AI-text och fokusdiagram.
- Sidstorlek, marginaler och sidnummer längst ned på varje sida.

Det här är ren layoutlogik – inga beräkningar, utan bara en beskrivning av hur rapporten ska se ut.

Val av kommun och grundläggande inställningar

Utifrån det valda kommunnamnet skapas en färgskala (`region_colors`) som:

- ger en färg åt “Alla kommuner (ovägt medel)”, och
- en annan färg åt fokuskommunen.

De här färgerna används senare i diagrammen, t.ex. för att rita fokuskommunen som en prick och jämförelsegruppen som trianglar.

Det skapas också en legendtext (förklaring) som återanvänds under alla diagram – där förklaras visuellt vad som är fokuskommunen och vad som är jämförelsevärden.

Inläsning av paket och externa funktioner

Scriptet ställer in generella inställningar för knitr (ingen kod ska visas i rapporten, inga varningar eller meddelanden i utskriften) och laddar sedan nödvändiga paket:

- *tidyverse* – datahantering och filtrering,
- *knitr* och *kableExtra* – tabeller,
- *Cairo* – grafik,
- *zoo* – stöd för tidsserier som lägger in dolda imputerade värden där data saknas i linjediagram
- *ellmer* –paket som underlättar kommunikation med AI-leverantörers API
- *readxl* – läsa Excel.

Dessutom läses en separat fil in: **Scriptfiler/rapportfunktioner.R**.

Den innehåller skräddarsydda hjälpfunktioner, till exempel:

- `generate_summary()` – skapar den övergripande AI-sammanfattningen,
- `generate_topic_comment()` – anropar AI-leverantörernas API och skapar korta AI-kommentarer per område,
- `create_focus_plot()` – ritar diagram för fokusnyckeltal,
- `get_border_color()` och `get_explanation_text()` – ger rätt färg och förklarande text till fokusdiagrammen.

Inläsning av färdiga datafiler

I nästa block läses datafilerna genererade av föregående script in:

- `rapport_data.rds` – huvuddatafilen med alla kommuner, nyckeltal, år, kvartiler, förändringar, indikatorer osv.

- rapport_data_rikt_medel.rds – motsvarande fil för Riket och “Alla kommuner (ovägt medel)”.
- top_muni_per_ntlD.rds – lista över kommuner som utmärker sig positivt per nyckeltal.
- comparegroups.rds – jämförelsegrupper baserade på befolkningsstorlek.

Därefter filtreras rapport_data så att endast rader som gäller den valda kommunen behålls. Denna filtrerade data kallas muni_data. Det är muni_data som nästan allt i rapporten bygger på.

Titel, datum, förord och signaturer

Scriptet läser sedan in informationen från excelfilen Definitioner.xlsx:

- Rapporttitel (t.ex. “Kommunrapport 2024”).
- Rapportdatum.
- Ett antal rader med förordstext, där varje stycke kan anges i en egen rad i excelfilen.

Därefter läses en annan flik med signaturer:

- Namn och titel på de personer som ska stå under förordet.

Scriptet bygger ihop alla dessa delar till:

- en banner överst med:
 - kommunnamn,
 - rapporttitel,
 - rapportdatum,
- ett förordsavsnitt i löpande text,
- ett block med signaturer (namn och titel) under förordet.

AI-genererad sammanfattning (exekutiv sammanfattning)

Efter förordet skapas ett avsnitt “Sammanfattning”. Där används funktionen generate_summary():

- den får in:
 - data för fokuskommunen (muni_data),
 - riksvärden och medelvärden (source_data_riks_medel),
 - kommunnamnet,

- den returnerar en AI-genererad text som sammanfattar läget och utvecklingen i kommunen.

Den här texten presenteras i en inramad ruta med etiketten “Texten är författad av en AI”.

Det blir rapportens ’exekutiva sammanfattning’, där de viktigaste budskapen lyfts fram på ett mer berättande sätt.

Avsnittet “Det här innehåller rapporten”

Direkt efter sammanfattningen följer en kort förklarande text om rapportens upplägg:

- att nyckeltalen är grupperade i olika områden,
- att varje område har en tabell med:
 - senaste tillgängliga värde,
 - förändring sedan föregående period,
- att bakgrundsfärgerna i tabellen visar:
 - hur kommunen ligger till jämfört med andra,
 - om utvecklingen gått i fördelaktig eller ofördelaktig riktning,
- och att ett diagram visas när vissa kriterier uppfylls.

Rapporten kör sedan en stor loop baserat på varje huvudrubrik som nyckeltalen är indelade under.

Huvudloopen: ett avsnitt per område

Rapportens huvuddel byggs genom en loop över alla områden (rubriker) som finns i `muni_data`. Dessa hämtas från variablerna `rubrik` och `rubrikordning`, så ordningen styrs av det data som skapats i förberedelsedescriptet.

För varje område sker följande:

1. En huvudrubrik sätts, t.ex. “Barn och unga”, “Ekonomi”, “Omsorg” osv.
2. Kort AI-kommentar om området
Funktionen `generate_topic_comment()` anropas. Den:
 - tittar på kommunens viktade nyckeltalsvärden inom just det här området,
 - jämför med kommunmedelvärdet,
 - formulerar en kort text om läget och utvecklingen inom området.
Texten visas i en särskild AI-märkt ruta, på samma sätt som sammanfattningen.

3. Scriptet plockar ut alla nyckeltal inom området för:

- den valda kommunen,
- det senaste tillgängliga året per nyckeltal.

4. Sedan skapas tabellkolumnerna:

- Nyckeltal (kortnamn/titel på nyckeltalet).
- År (det år värdet avser – kan skilja mellan nyckeltal).
- Värde:
 - talet formateras med rätt antal decimaler,
 - decimaltecken och tusentalsavgränsare anpassas för svensk stil.
- Förändring:
 - visar förändringen sedan föregående år,
 - om jämförelsen i stället görs mot två år bakåt markeras värdet med en asterisk (*),
 - ett plus-tecken sätts ut vid positiva förändringar.

5. Färgkodning i tabellen - två kolumner får bakgrundsfärg:

- Värdekolumnen (nivåjämförelse)
 - Grön bakgrund om kommunen tillhör de mest fördelaktiga 25 % (beroende på om högt eller lågt är bra).
 - Röd bakgrund om kommunen tillhör de mest ofördelaktiga 25 %.
 - Gul bakgrund för mellangruppen.
 - Vit bakgrund om nyckeltalet inte har definierats som “högt är bra” eller “lågt är bra” (Lowgood = 2 eller saknas).
- Förändringskolumnen (förändringsriktning)
 - Grön bakgrund om utvecklingen gått i fördelaktig riktning givet hur nyckeltalet tolkas.
 - Röd bakgrund om utvecklingen gått i ofördelaktig riktning.
 - Vit bakgrund om förändringen är noll eller ingen önskvärd riktning är definierad.

Tabellen grupperas också efter underrubrik där sådana finns, så att relaterade nyckeltal hamnar tillsammans under en egen mellanrubrik.

6. Om någon förändring i tabellen är beräknad mot året två år bakåt i stället för föregående år, läggs en fotnot till under tabellen som förklarar detta.

Fokusnyckeltal inom varje område

Efter tabellen gör scriptet en särskild genomgång av fokusnyckeltal inom området. Här används den information som tidigare skapades i scriptet som hämtade all data.

- rankning av de fem mest fördelaktiga och fem mest ofördelaktiga förändringarna,
- markering av nyckeltal som länge legat på en stabilt hög eller stabilt låg nivå (fåglar/fiskar).

För varje område sker följande:

1. Scriptet plockar ut de rader där kommunen:
 - har någon av fokusrankerna (bäst/sämst förändring), eller
 - har etiketten “fish” eller “bird”.
2. Om det inte finns några sådana nyckeltal:
 - skapas ett kort textstycke som förklarar att området inte innehåller nyckeltal som uppfyller kriterierna för att räknas som fokusnyckeltal,
 - därefter går scriptet vidare till nästa område.
3. Om det finns fokusnyckeltal:
 - väljs upp till åtta nyckeltal ut, med en prioritering där:
 - nyckeltal med fish/bird-indikatorer prioriteras först,
 - därefter de med högst fokusrankning.
 - En inledande text för avsnittet förklarar vad som visas:
 - tydliga förbättringar/försämringar,
 - och/eller nyckeltal som legat kvar på en mycket hög eller låg nivå över tid.

Diagram och förklarande texter för fokusnyckeltal

För de utvalda fokusnyckeltalen skapas linjediagram som visar utvecklingen över tid:

- Diagrammen ritas med funktionen `create_focus_plot()`:
 - fokuskommunen markeras tydligt,
 - jämförelser mot riket/medel och/eller jämförelsegrupper kan ingå (beroende på hur funktionen är skriven),
 - en gemensam legend visar vad som är vad (triangel vs cirkel).
- Varje diagram placeras i en inramad ruta ("rounded-box") där:
 - rubriken är nyckeltalets namn,
 - diagrammet visas centrerat,
 - en tillhörande, regelstyrd förklarande text läggs under diagrammet.

Den förklarande texten tas fram av funktionen `get_explanation_text()` som:

- tittar på:
 - om utvecklingen varit fördelaktig eller ofördelaktig,
 - om nyckeltalet är en "fisk" (stabilt ofördelaktigt) eller "fågel" (stabilt fördelaktigt),
 - hur kommunen ligger till jämfört med sina jämförelsekommuner (via `compare_groups`),
 - och listan med föredömliga kommuner per nyckeltal (`top_muni_per_ntID`),
- och formulerar en kort, tolkningshjälpande text under diagrammet.

Antalet diagram per sida anpassas:

- 1 eller 2 nyckeltal → en eller två större rutor på sidan.
- 3–8 nyckeltal → grupperas mindre i 2×2-rutor över flera sidor.

Avsnittet “Om rapporten”

I slutet av rapporten finns ett fast avsnitt “Om rapporten” där tolkningen av tabeller och fokusnyckeltal beskrivs mer systematiskt.

Här förklaras bland annat:

- hur tabellerna är uppbyggda,
- vad färgerna (grön, gul, röd) betyder i både värde- och förändringskolumn,
- att en asterisk (*) betyder jämförelse med år T–2 i stället för år T–1,
- vad fokusnyckeltal är,
- vilka kriterier som används för att:
 - välja ut de största förbättringarna/försämringarna,
 - avgöra om ett nyckeltal är “stabilt fördelaktigt” eller “stabilt ofördelaktigt”,
- och hur ramen runt fokusdiagrammen ska tolkas:
 - grön ram – tillhör de mest fördelaktiga förändringarna eller långvarigt god nivå,
 - röd ram – långvarigt eller tydligt ofördelaktig utveckling,
 - grå ram – blandad signal (t.ex. hög nivå men försämrad utveckling).

Det finns också en enkel färgruta-illustration (små färgfält) som visar färgskalorna visuellt.

Läget – Generate reports.R

Scriptet är ett “masskörnings-script” som automatiskt skapar rapporter för flera kommuner.

- Det läser in vilka kommuner som ska ingå från Definitioner.xlsx (fliken *Kommuner*, där *Inkludera* = 1).
- För varje kommun kör det rapportmallen “Läget - Generera rapport.Rmd” och sparar resultatet som HTML i mappen *Rapporter*.
- Därefter konverteras varje HTML-rapport till en PDF (också i *Rapporter*), med filnamn som innehåller kommun och år.
- Till sist skriver scriptet ut hur många rapporter som skapades och hur lång tid hela körningen tog. Att generera 290 rapporter tar drygt 6 timmar.