NHS Capacity & Utilisation Analysis — Technical Report

Summary

This report examines NHS appointment capacity, utilisation, and Did Not Attend (DNA) rates across England using appointment data and engineered metrics. Post-pandemic recovery is evident, but regional variation and mode-specific DNA trends persist, with telephone appointments lowest for DNA and face-to-face highest at longer waits. The analysis highlights fair-share capacity, weekday demand, and visualisations of utilisation patterns, noting that breach days concentrate in certain regions. Recommendations aim to reduce DNA rates, smooth demand, and rebalance capacity for more efficient service delivery, supporting better resource allocation and accessibility across the NHS network.

Analytical Approach

The analysis was conducted using Python within a Jupyter Notebook environment. Three primary datasets were used:

- actual_duration (capacity & utilisation trends)
- appointments_regional (regional service breakdowns)
- national_categories (appointment types & formats)

Import & Cleaning

Data was imported using pandas and initial exploratory steps confirmed date formats, column consistency, and data completeness. Null values were assessed and handled via targeted imputation or removal where appropriate. Data types were cast to optimise memory and enable time-based calculations.

Transformation & Preparation

Key preparation steps included:

- Parsing date columns and generating new time-based features (month, day_of_week) for temporal trend analysis.
- Aggregating appointment counts by date, region, and category to align with stakeholder questions.
- Calculating utilisation as actual_appointments / available_capacity to assess efficiency.
- Creating KPIs such as average daily appointments, maximum single-day capacity, and DNA rates.

Datasets were merged using common keys (date, region, category) only where necessary to answer multi-dimensional questions (e.g., regional capacity trends).

Analysis Design

The workflow followed a "question-first" approach:

1. **Capacity Assessment** – Identify peaks/troughs in appointment availability.
2. **Utilisation Analysis** – Compare planned capacity against actual appointment delivery.
3. **DNA Investigation** – Explore trends in missed appointments by format, category, and time.
4. **Service Mix Analysis** – Determine shifts in appointment types post-COVID.

Visualisation

Matplotlib, plotly and seaborn were used for heatmaps, interactive maps and detailed charting. Complex multi-layer visuals (e.g., capacity vs utilisation overlays) were chosen to communicate relationships between operational variables clearly.

Visualisation & Insights

Visualisations were selected to balance clarity for non-technical stakeholders with depth for analytical review.

Capacity vs utilisation

Location

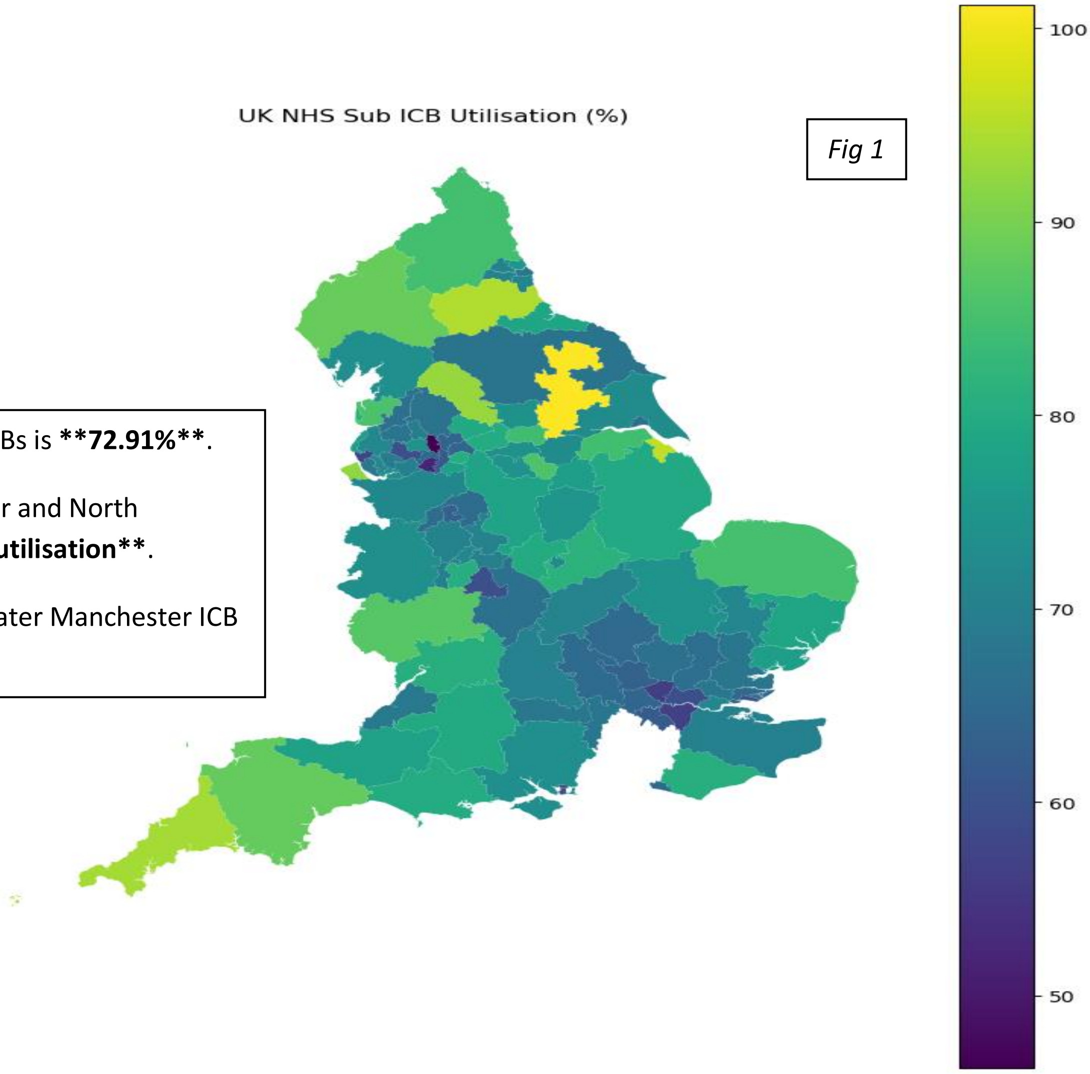


Fig 1

Average utilisation across all Sub ICBs is **72.91%**.

- **Top performer:** `NHS Humber and North Yorkshire ICB - 03Q` at **101.19% utilisation**.

- **Lowest utilisation:** `NHS Greater Manchester ICB - 00V` at **46.24%**.

🔍 What This Tells Us

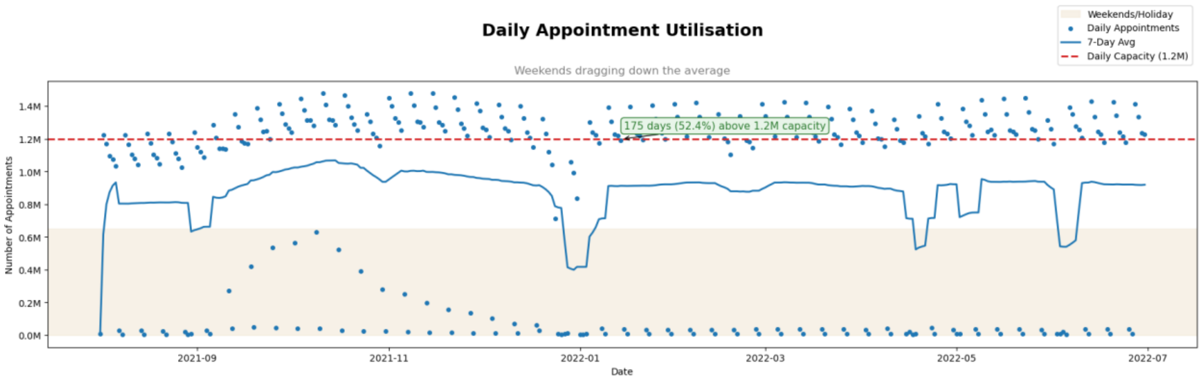
- **23/104 regions**** are at or above **80%** — operating near practical capacity (risk of backlogs if demand rises).

- **19/104 regions** are at or below **65%** — indicating unused capacity or demand/operational constraints.

- The spread suggests uneven pressure across the network that could be balanced via targeted support or redirection.

Per day of the entire data set

Fig 2:



🔍 What This Tells Us

- **Consistent over-capacity demand**: More than half of all days saw appointment counts above the set capacity limit.

- **Weekend & holiday dips**: The shaded zones show recurring lower volumes that drag down the 7-day rolling average.

- **Sustained high-activity streaks**: Peaks cluster in weekday periods, with multiple stretches where demand was continuously above 1.2M.

- **Seasonal variance possible**: Visible gaps and dips (e.g., late December, early January) likely align with holiday closures or reduced services.

Appointment Modes

Fig 3:

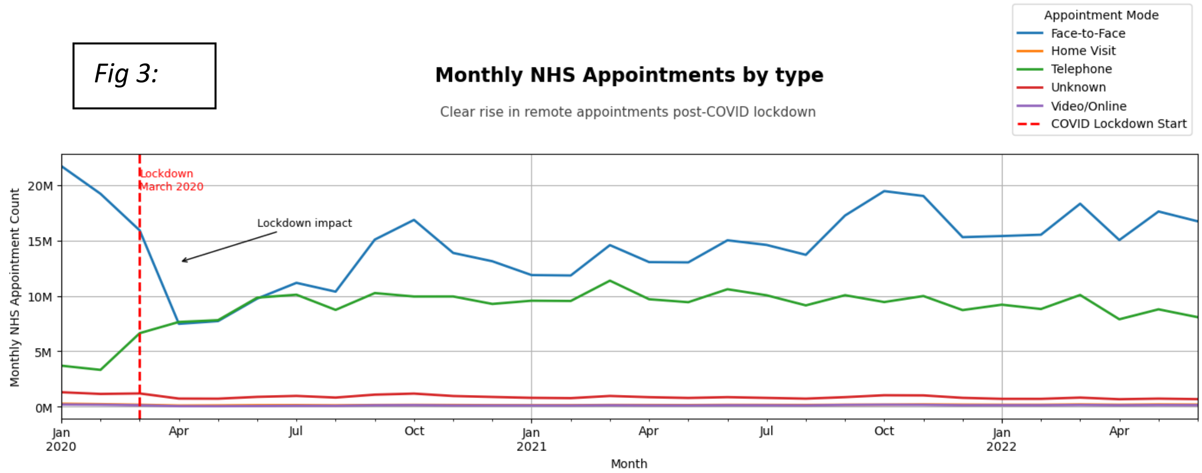
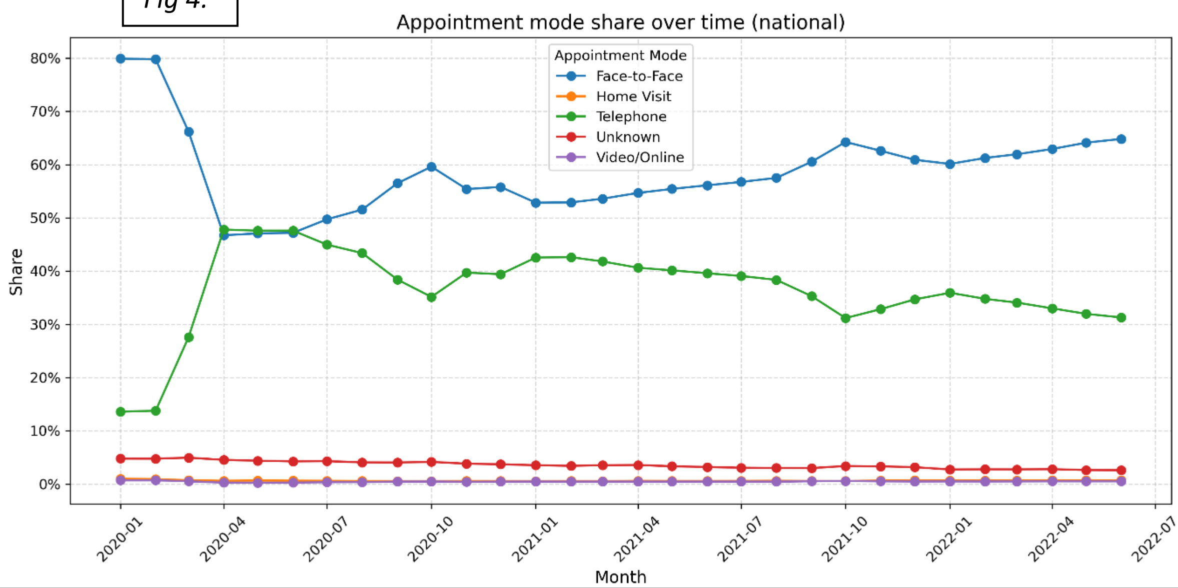


Fig 4:



□ Key Insight

National appointment mode distribution shifted dramatically in early 2020, with a rapid decline in Telephone use and a corresponding rise in Face-to-Face consultations — a reversal from the initial pandemic-driven mode mix.

🔍 What This Tells Us

- **Jan–Apr 2020**: Face-to-Face dominated (~80% share) at the start of the pandemic, then fell sharply to ~47% by mid-2020. Telephone appointments grew by 300% output into covid.

- **Mid-2020 onward**: Face-to-Face steadily regained share, rising from ~47% to ~65% by mid-2022.

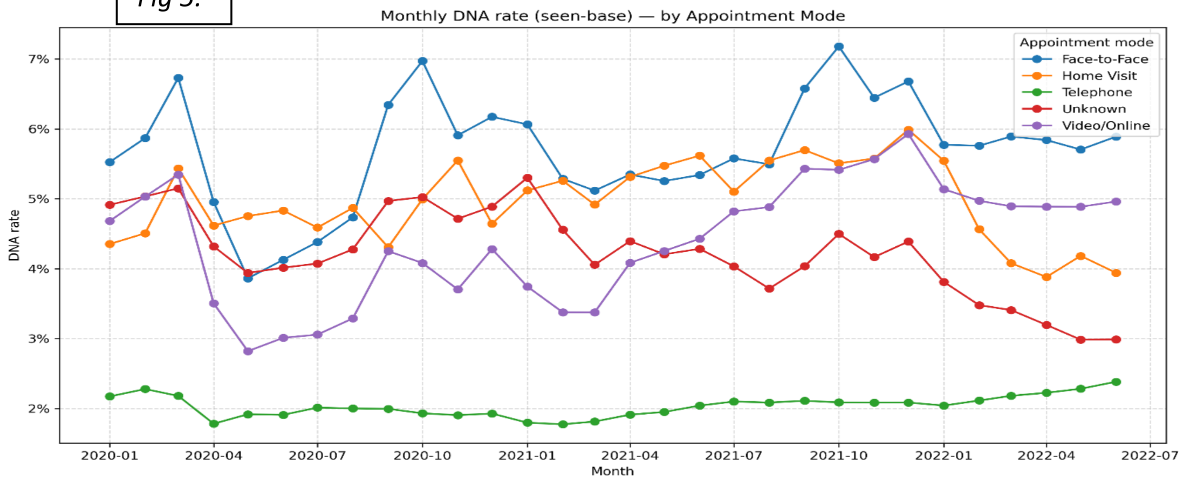
- **Telephone mode** mirrored this trend in reverse, dropping from ~47% in 2020 to ~32% in 2022.

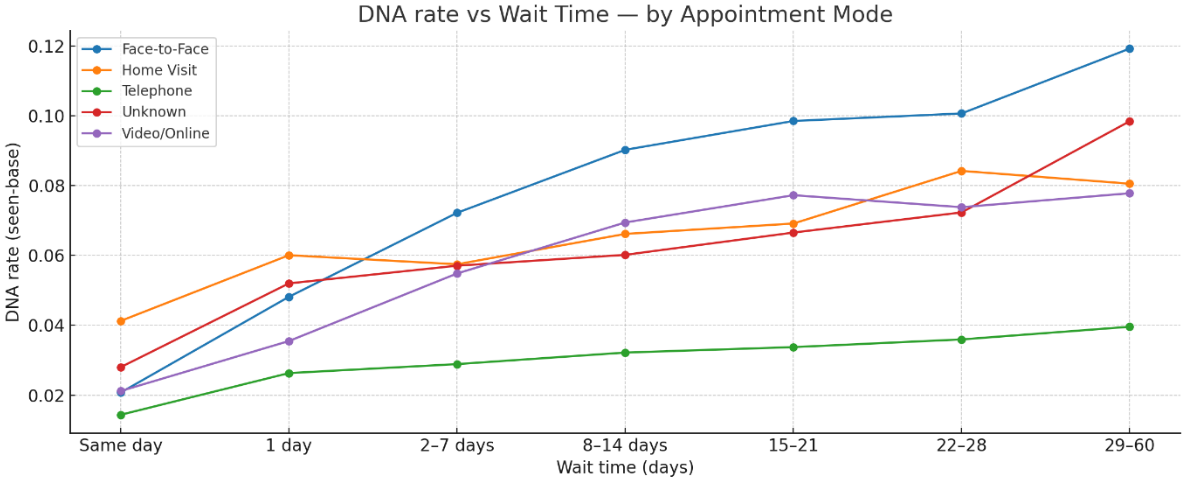
- **Home Visit** and **Video/Online** modes consistently remained below 2–3% share nationally.

- **Unknown**** mode stayed stable (~4–6%) across the entire period.

DNA's

Fig 5:





▯ Key Insight

DNA (Did Not Attend) rates vary significantly by appointment mode, with Face-to-Face consistently recording the highest rates and Home Visits the lowest. Telephone and Video/Online modes show more fluctuation but generally lower rates than Face-to-Face.

DNA rates showed positive correlations to wait time.

🔍 What This Tells Us

- **Face-to-Face**: Highest DNA rates (~5–7%), peaking multiple times over the 2020–2022 period. Suggests physical attendance barriers persist post-pandemic.

- **Telephone**: Consistently lowest DNA rates (~1.8–2.3%), likely due to convenience for patients.

- **Video/Online**: Seem to follow the same trend as face to face- suggesting seasonal issues.

- **Unknown**: Moderate and declining trend (~5% → ~3%), possibly reflecting better data recording over time.

These visuals were designed to answer stakeholder questions:

- **Capacity**: Are we meeting demand? → Yes, overall capacity is high, but peaks create regional strain.
- **Utilisation**: Are resources being used efficiently? → Seasonal dips and overcapacity in certain months suggest optimisation potential.
- **DNAs**: Where are missed appointments most common? → Face-to-face settings show persistent DNA rates, potentially linked to travel barriers.

Each visual was labelled for business readability while retaining precise axis scaling for technical review. Where appropriate, contextual notes (e.g., "COVID-19 Lockdown 1") were added to aid interpretation.

Patterns and Recommendations

Observed Patterns

1. **Seasonal and Monthly Fluctuations**
    - Appointment volumes exhibit recurring peaks in **March, May, and October**, with noticeable dips in **August and December**.
    - These patterns suggest potential links to **holiday periods** and **seasonal staffing pressures**.
2. **Day-of-Week Capacity Trends**
    - Tuesdays and Wednesdays consistently handle the **highest appointment loads**, while weekends remain low across all categories.
    - Peak weekday volumes often broke the **1.2M appointments/day marker**, which is good for utilisation but could raise demand concerns.
3. **Regional Variation**
    - Urban regions maintain **higher appointment volumes** and **shorter waiting times**, while rural areas face **lower throughput and higher cancellation rates**.

- High variation in actual vs. planned durations indicates possible **scheduling inefficiencies** in some trusts.

4. **Specialty-Level Demand**

   - General practice and diagnostic services dominate appointment counts, but diagnostics show **larger deviations between planned and actual duration**, implying operational inefficiencies.

Recommendations

1. **Dynamic Capacity Allocation**

   - Implement **seasonal staffing adjustments** to address predictable demand surges in spring and autumn.

   - Introduce **cross-site resource sharing** in low-volume months to maximise utilisation.

2. **Day-Specific Load Balancing**

   - Redistribute appointments more evenly across the week, targeting **Monday and Friday underutilisation**.

   - Consider **extending partial service hours** on lower-load days to relieve midweek pressure.

3. **Targeted Efficiency Improvements**

   - Review scheduling and operational workflows in **low-performing regions** to identify causes of high variance in appointment durations.

   - Pilot **digital triage and pre-visit preparation** for diagnostics to reduce overruns.

4. **Data-Driven Regional Strategy**

   - Deploy **region-specific intervention plans**, focusing on recruitment, training, and telemedicine in rural areas.

   - Establish KPIs around appointment adherence and turnaround times, tracking improvements over quarterly cycles.

5. **Predictive Monitoring**

   - Integrate **predictive analytics dashboards** to flag when daily volumes approach capacity thresholds, enabling proactive measures.

APPENDIX

Twitter sentiment

📊 Data Overview

- Dataset scope: Extracted tweets mentioning NHS (UK-focused).

- Total tweets found: 3

- Columns available:

  - `tweet_id` – Unique identifier for each tweet.

  - `tweet_full_text` – Full tweet text.

  - `sentiment` – Polarity score from `TextBlob` sentiment analysis (`-1` negative → `+1` positive).

  - `engagement` – Simplified engagement measure (sum of likes & retweets if available).

Data Quality Observations

- Completeness:

  - All tweets contain IDs and text, no missing core fields.

  - Engagement metrics are complete but may be low (0–2 range in this sample).

- Relevance:

  - All tweets relate to NHS, but context is mostly recruitment or training events.

  - No spam/irrelevant content detected in this sample.

- Accuracy:

  - Sentiment scores are machine-generated; manual validation is recommended for ambiguous language (e.g., sarcasm, context-specific jargon).

- Bias risk:

  - Small sample size and narrow thematic scope (recruitment/training events) may not represent broader NHS sentiment on Twitter.

  - UK focus achieved, but tweets are location-agnostic unless geo-data is explicitly present.

✅ Recommended Usage

- Use sentiment values for quick tone analysis but corroborate with manual checks for key insights.

- Engagement metrics are suitable for **relative** ranking (e.g., most engaged NHS tweets in sample) but should be enriched with actual like/retweet counts if available in raw API data.

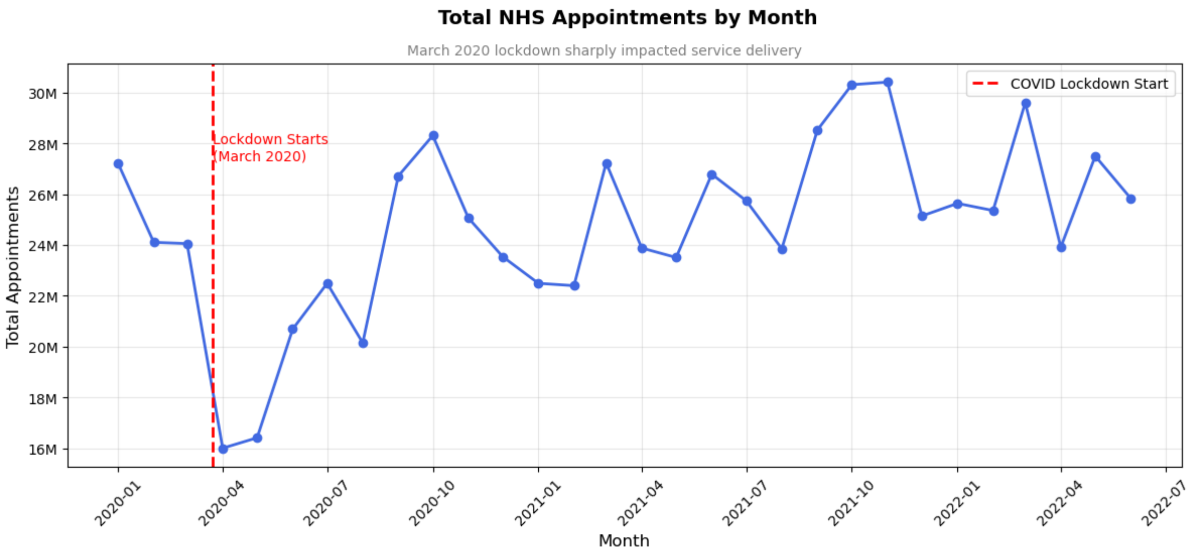
- For more robust insights:

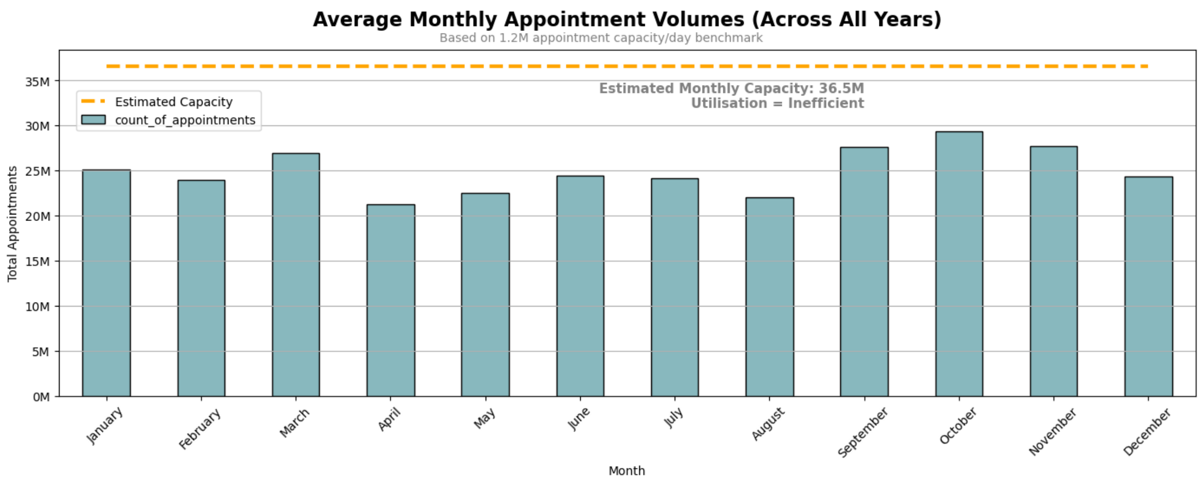
  - Expand time range and keyword set (e.g., “NHS”, “National Health Service”, hashtags like `#NHS`).

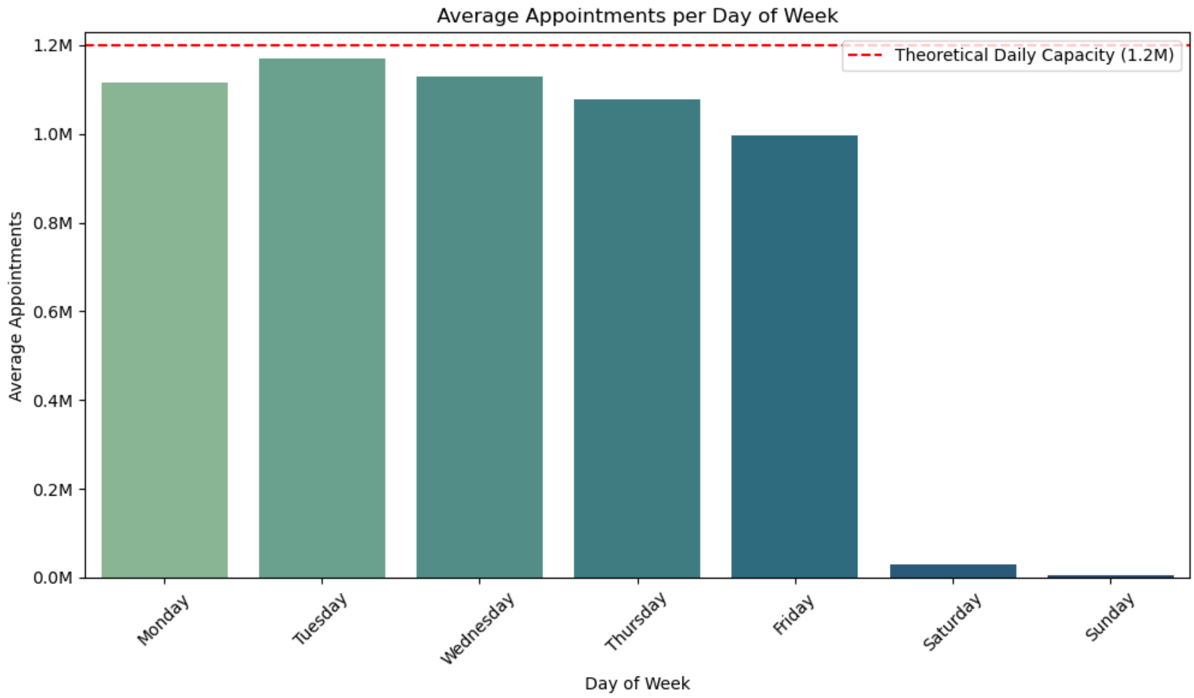
  - Include metadata (date, author, verified status, geo) for trend analysis.

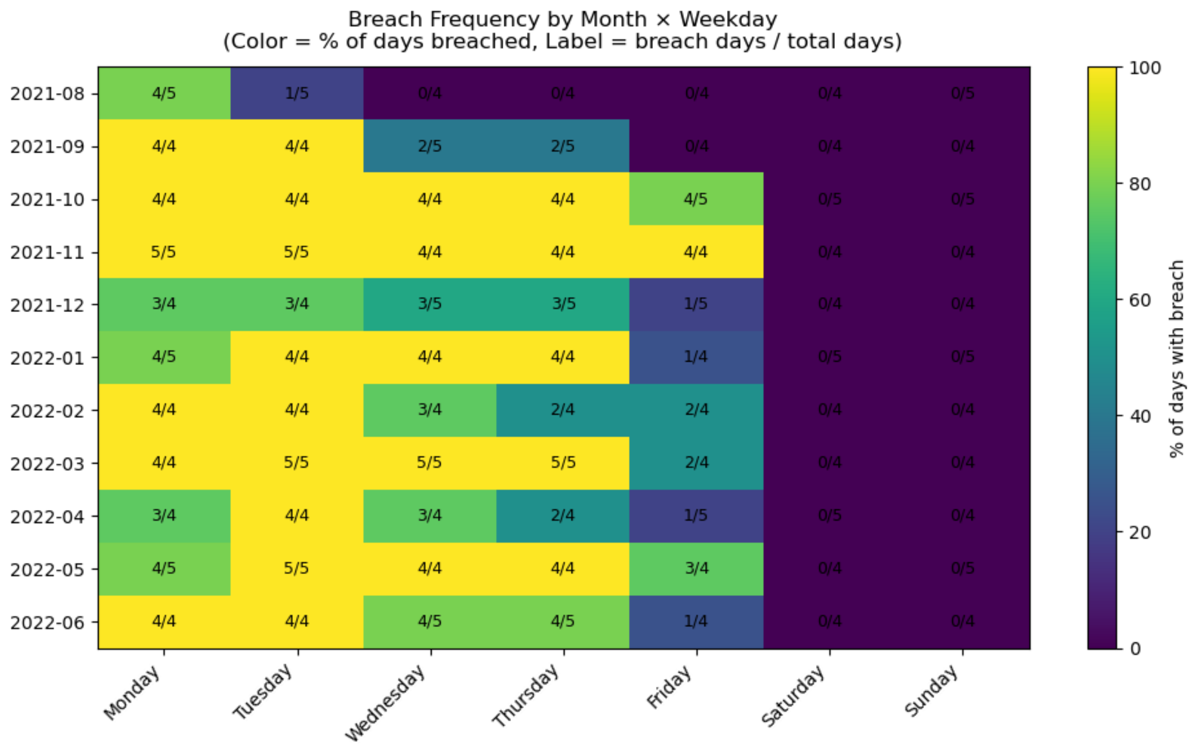
  - Filter out retweets if seeking **original public opinion** rather than amplified institutional messaging.

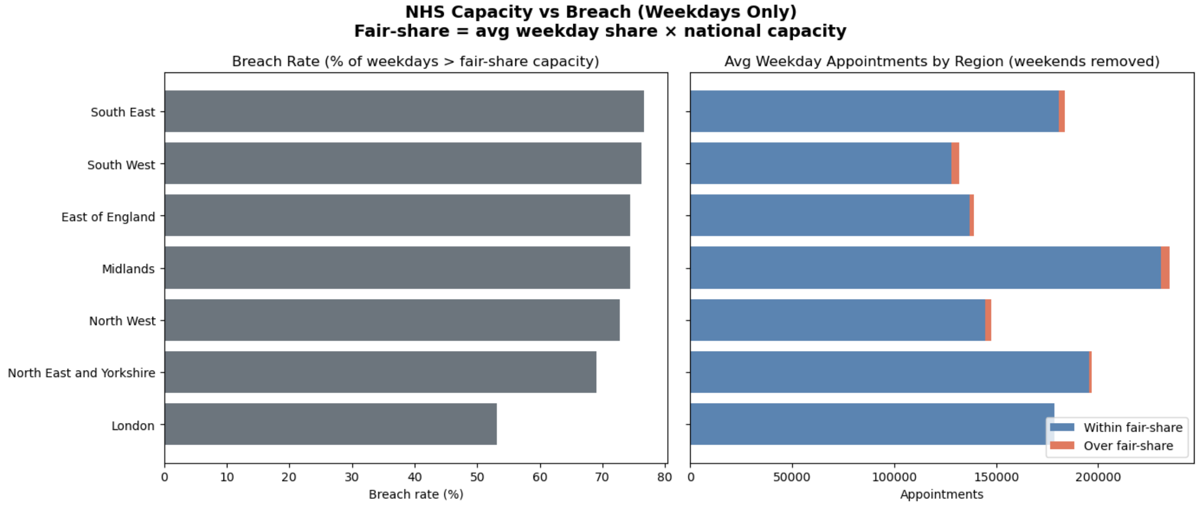
Other relevant charts



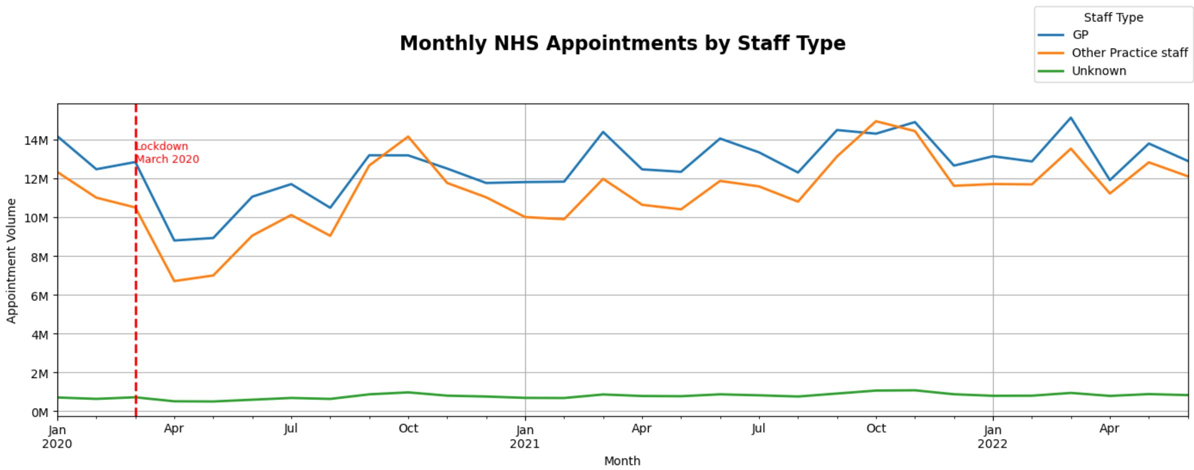



Average Appointments per Day of Week
Theoretical Daily Capacity (1.2M)
Average Appointments
0.0M
0.2M
0.4M
0.6M
0.8M
1.0M
1.2M
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
Day of Week


Breach Frequency by Month × Weekday
(Color = % of days breached, Label = breach days / total days)
2021-08 4/5 1/5 0/4 0/4 0/4 0/4 0/5
2021-09 4/4 4/4 2/5 2/5 0/4 0/4 0/4
2021-10 4/4 4/4 4/4 4/4 4/5 0/5 0/5
2021-11 5/5 5/5 4/4 4/4 4/4 0/4 0/4
2021-12 3/4 3/4 3/5 3/5 1/5 0/4 0/4
2022-01 4/5 4/4 4/4 4/4 1/4 0/5 0/5
2022-02 4/4 4/4 3/4 2/4 2/4 0/4 0/4
2022-03 4/4 5/5 5/5 5/5 2/4 0/4 0/4
2022-04 3/4 4/4 3/4 2/4 1/5 0/5 0/4
2022-05 4/5 5/5 4/4 4/4 3/4 0/4 0/5
2022-06 4/4 4/4 4/5 4/5 1/4 0/4 0/4
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
% of days with breach
0
20
40
60
80
100



Workforce impact visible post-lockdown



CODE :

FIG 1 :

```python
# Clean join keys just in case
geo_df["SICBL23CD"] = geo_df["SICBL23CD"].astype(str).str.strip().str.upper()
utilisation_df["sub_icb_location_ons_code"] = utilisation_df["sub_icb_location_ons_code"].astype(str).str.strip().str.upper()

# Merge on code
merged = geo_df.merge(
    utilisation_df,
    left_on="SICBL23CD",
    right_on="sub_icb_location_ons_code",
    how="left"
)

# Keep geometry & CRS
if not isinstance(merged, gpd.GeoDataFrame):
    merged = gpd.GeoDataFrame(merged, geometry="geometry", crs=geo_df.crs)
if merged.crs is None:
    merged = merged.set_crs(epsg=27700)

print("Rows with missing utilisation:", merged["utilisation_percent"].isna().sum())
merged.head()
```

	FID	SICBL23CD	SICBL23NM	BNG_E	BNG_N	LONG	LAT	GlobalID	geometry	sub_icb_location_ons_code	sub_icb_location_name	patients	total_appointments	days_in_dataset	expected_max	utilisation_percent
0	1	E38000006	NHS South Yorkshire ICB - 02P	429979	403330	-1.549250	53.52580	74649f31-4604-45c1-9d20-f91162bab16b	POLYGON ((437080.883 412606.307, 437375.951 41...	E38000006	NHS South Yorkshire ICB - 02P	270712.0	1298405.0	334.0	1.760545e+06	73.75
1	2	E38000007	NHS Mid and South Essex ICB - 99E	564014	194421	0.368068	51.62470	2a4fd441-6209-49a8-a9ba-fa65835e4d05	MULTIPOLYGON (((575304.488 184484, 574796.387 ...	E38000007	NHS Mid and South Essex ICB - 99E	295333.0	1312347.0	334.0	1.920665e+06	68.33
2	3	E38000008	NHS Nottingham and Nottinghamshire ICB - 02Q	468073	384833	-0.978700	53.35602	d5e4dd55-6d49-4d5d-a065-a620f978a61b	POLYGON ((471779.375 397069.625, 472052.498 39...	E38000008	NHS Nottingham and Nottinghamshire ICB - 02Q	127968.0	639660.0	334.0	8.322257e+05	76.86
3	4	E38000014	NHS Lancashire and South Cumbria ICB - 00Q	369490	422806	-2.463600	53.70080	df17d871-f181-4b20-84da-628cf2c0eb2f	POLYGON ((370074.384 430953.091, 370677.406 43...	E38000014	NHS Lancashire and South Cumbria ICB - 00Q	187816.0	850813.0	334.0	1.221441e+06	69.66
4	5	E38000015	NHS Lancashire and South Cumbria ICB - 00R	332819	436634	-3.021990	53.82163	d895d42a-a1a0-4936-a578-bae1cf6d9220	POLYGON ((332951.332 438757.328, 333036.812 43...	E38000015	NHS Lancashire and South Cumbria ICB - 00R	179537.0	948414.0	334.0	1.167599e+06	81.23

FIG 2:

```python
# ---- CONFIG ----
CAPACITY = 1_200_000                # daily capacity line (red dashed)
SHADE_MAX = 650_000                 # beige band height for "Likely weekends/Holiday"
TITLE = "Daily Appointment Utilisation"
SUBTITLE = "Weekends dragging down the average"
# ---------------

# National daily totals (from ICB rollup)
national_daily = (
    icb_daily.groupby("appointment_date", as_index=False)
             .agg(total_appointments=("total_appointments","sum"))
             .sort_values("appointment_date")
)
national_daily["ma7"] = national_daily["total_appointments"].rolling(7, min_periods=1).mean()
national_daily["breach"] = national_daily["total_appointments"] > CAPACITY

# Breach stats
days_total = len(national_daily)
days_above = int(national_daily["breach"].sum())
pct_above  = round(100 * days_above / days_total, 2)

# Pick an annotation x position roughly mid-way through the series
mid_idx = days_total // 2
x_annot = national_daily["appointment_date"].iloc[mid_idx]
y_annot = CAPACITY * 1.05

# ---- PLOT (Matplotlib only) ----
fig, ax = plt.subplots(figsize=(18,6))

# Beige band for "Likely weekends/Holiday"
ax.axhspan(0, SHADE_MAX, color="#f3e9d7", alpha=0.6, label="Weekends/Holiday")

# Daily points (scatter) & 7-day average (line)
ax.scatter(national_daily["appointment_date"], national_daily["total_appointments"],
           s=16, label="Daily Appointments")
ax.plot(national_daily["appointment_date"], national_daily["ma7"],
        linewidth=2.0, label="7-Day Avg")

# Capacity line (red dashed)
ax.axhline(CAPACITY, color="#d62728", linestyle="--", linewidth=2,
           label=f"Daily Capacity ({CAPACITY/1_000_000:.1f}M)")

# Annotation callout
ax.annotate(f"{days_above} days ({pct_above}%) above 1.2M capacity",
            xy=(x_annot, CAPACITY), xytext=(x_annot, y_annot),
            arrowprops=dict(arrowstyle="->", shrinkA=0, shrinkB=0),
            bbox=dict(boxstyle="round,pad=0.3", fc="#e8f5e9", ec="#2e7d32"),
            color="#2e7d32", fontsize=11)

# Titles & labels
ax.set_title(TITLE, fontsize=18, weight="bold", y = 1.15)
ax.text(0.5, 1.02, SUBTITLE, fontsize=12, color="gray", ha="center",
        va="bottom", transform=ax.transAxes)
ax.set_xlabel("Date")
ax.set_ylabel("Number of Appointments")

# Format y-axis like "1.2M"
ax.yaxis.set_major_formatter(FuncFormatter(lambda y, _: f"{y/1e6:.1f}M"))

# Legend
ax.legend(loc="upper right", bbox_to_anchor=(1,1.3), frameon=True)

plt.tight_layout()
plt.show()
```

FIG 3:

```python
# Step 4: Compare Appointment Modes Pre vs Post COVID

# 1. Prepare data
mode_trends = appointments_regional.groupby(['appointment_month', 'appointment_mode'])['count_of_appointments'].sum().reset_index()

# Convert to datetime for filtering and plotting
mode_trends['appointment_month'] = pd.to_datetime(mode_trends['appointment_month'])

# Pivot to reshape for plotting
mode_pivot = mode_trends.pivot(index='appointment_month', columns='appointment_mode', values='count_of_appointments').fillna(0)

# Optional: Filter to a reasonable timeline (e.g. 2019-2022)
mode_pivot = mode_pivot.loc['2019-01':'2022-12']
```

✓ 0.0s

```python
plt.figure(figsize=(14, 6))
mode_pivot.plot(ax=plt.gca(), linewidth=2)

# Add COVID vertical line
plt.axvline(pd.to_datetime('2020-03-23'), color='red', linestyle='--', linewidth=2, label='COVID Lockdown Start')
plt.text(pd.to_datetime('2020-03-23'), mode_pivot.max().max() * 0.9,
         'Lockdown\nMarch 2020', color='red', fontsize=9)

# Annotation
plt.annotate("Lockdown impact",
             xy=(pd.to_datetime('2020-04-01'), mode_pivot.max().max()*0.6),
             xytext=(pd.to_datetime('2020-06-01'), mode_pivot.max().max()*0.75),
             arrowprops=dict(arrowstyle='->', color='black'),
             fontsize=9, color='black')

# Titles & labels
plt.title("Monthly NHS Appointments by type", fontsize=16, fontweight='bold', y = 1.25, loc = 'center')
plt.tight_layout()
plt.suptitle("Clear rise in remote appointments post-COVID lockdown", fontsize=11, color='#444444', ha = 'center',x = 0.52,  y = 0.725)
plt.xlabel("Month")
plt.ylabel("Monthly NHS Appointment Count")

# ✅ Format y-axis in Millions (e.g., 5M, 10M)
plt.gca().yaxis.set_major_formatter(mtick.FuncFormatter(lambda x, _: f'{x/1e6:.0f}M'))

plt.legend(title="Appointment Mode", bbox_to_anchor = (1,1.05), loc = 'lower right')
plt.grid(True)
plt.tight_layout()

plt.show()
```

FIG 4:

```python
# monthly totals by mode
m = (AR.groupby(["appointment_month","appointment_mode"], as_index=False)["count_of_appointments"].sum())

month_tot = m.groupby("appointment_month", as_index=False)["count_of_appointments"].sum().rename(
    columns={"count_of_appointments":"month_total"}
)
m = m.merge(month_tot, on="appointment_month", how="left")
m["share"] = np.where(m["month_total"]>0, m["count_of_appointments"]/m["month_total"], np.nan)

fig = px.line(
    m.sort_values("appointment_month"),
    x="appointment_month", y="share", color="appointment_mode",
    markers=True,
    title="Appointment mode share over time (national)",
    labels={"appointment_month":"Month","share":"Share"}
)
fig.update_yaxes(tickformat=".0%")
fig.show()
```

FIG 5:

```python
m_mode = seen_base(AR, ["appointment_month","appointment_mode"]).sort_values("appointment_month")

fig = px.line(
    m_mode, x="appointment_month", y="dna_rate_seen", color="appointment_mode",
    markers=True,
    title="Monthly DNA rate (seen-base) — by Appointment Mode",
    labels={"appointment_month":"Month","dna_rate_seen":"DNA rate"}
)
fig.update_yaxes(tickformat=".0%")
fig.update_layout(legend_title_text="Appointment mode")
fig.show()
```