Cracking the Code of Champions:
What the Data Reveals About Top-10 Tennis Players

🟦 Business Task

I investigated the performance differences between the world's top ATP players and lower-ranked players. The goal is to uncover which match statistics most consistently separate top-10 players from the rest of the field — and how these differences vary across court surfaces (grass, clay, and hard).

These insights can be used to inform coaches, training academies, and player development specialists about which performance areas to prioritize for each surface type.

🟦 Stakeholders

- **Coaches and training staff** looking to structure surface-specific training plans
- **Player development academies** focusing on long-term progression
- **Sports analysts and journalists** interested in performance trends
- **Aspiring professional players** aiming to model top-performer habits

---

🟦 Key Questions

- Which match statistics most clearly distinguish top-10 players from lower-ranked players?
- How do these key statistics vary across different surface types (grass, clay, hard)?
- What actionable insights can inform training and development strategies for rising players?

🟦 Metrics of interest

- First and second serve win %
- Return points won
- Break point conversion
- Winners vs. unforced errors

- Net approaches (if available)
- Match duration
- Rally length
- Surface type
- ATP ranking

🟦 Dataset

This case study uses the [ATP Tennis Dataset from Kaggle] containing **188,162 men's matches** from **1968 to 2022**. Each row includes detailed metadata, player rankings, and performance statistics.

For this project, I focused on **Grand Slam matches (tourney_level = 'G')** from **2000–2022**, where complete winner statistics were available.

Key features include:

- **Match metadata:** tournament name, surface, round, date (tourney_date)
- **Player info:** winner/loser names and ATP rankings at the time (winner_rank, loser_rank)
- **Performance metrics:**
    - Serve: aces, double faults, first/second serve points won
    - Pressure points: break points faced/saved
- **Surface type:** surface (Hard, Clay, Grass)
- **Tournament level:** filtered to Grand Slams only (tourney_level = 'G')

This rich dataset enabled me to compare **serve-related performance** between ATP top-10 players and other ranking groups, with breakdowns per **surface type**.

🟦 Data Preparation and cleaning

To manage the large dataset (~188,000 rows), I used **Google BigQuery** for querying and **Excel** for visualization. The dataset required significant cleaning before meaningful analysis could begin. Key steps included:

🔧 In BigQuery:

- **Filtered** to include only **Grand Slam matches** (using tourney_level = 'G') from **2000–2022**
- **Removed rows with missing values** in critical serve-related stats (e.g., w_svpt, w_1stWon, w_df, etc.)
- **Corrected data types**: many numeric stats (like w_ace, w_bpFaced) were incorrectly stored as strings → converted using SAFE_CAST
- **Handled divide-by-zero cases** using NULLIF to avoid errors when calculating percentages
- **Created new ranking categories** using a CASE statement to bucket players into Top 10, Top 50, Top 100, and Below 100
- **Aggregated results** by ranking group and surface to prepare for visualization
- **See the appendix** for the full SQL queries.

🧼 In Excel:

- After exporting, **Excel treated numeric columns as text** due to locale settings → resolved with Text to Columns and formatting fixes
- **Added a helper column** to manually define ranking order (to control sort order in PivotTables)
- **Verified all numeric columns** using ISNUMBER() and adjusted formatting to enable correct filtering, sorting, and charting

---

🟦 Analysis

To compare performance across player levels and surfaces, I grouped the match data by:

- **Winner's ranking group**:
  - *Top 10*
  - *Top 11–50*
  - *Top 51–100*
  - *Below 100*
- **Surface type**: Hard, Clay, and Grass

From there, I calculated and analyzed the following key performance metrics:

- Average **aces** per match
- Average **double faults** per match
- **First serve percentage**
- **First serve win percentage**
- **Second serve win percentage**
- **Break point save percentage**

These stats were calculated **only for match winners**, to ensure fair comparison and reduce variability caused by short or unrepresentative losses.

The analysis focused on identifying which of these metrics most clearly distinguish top-10 players — and how those distinctions shift by surface.

🟦 Insights

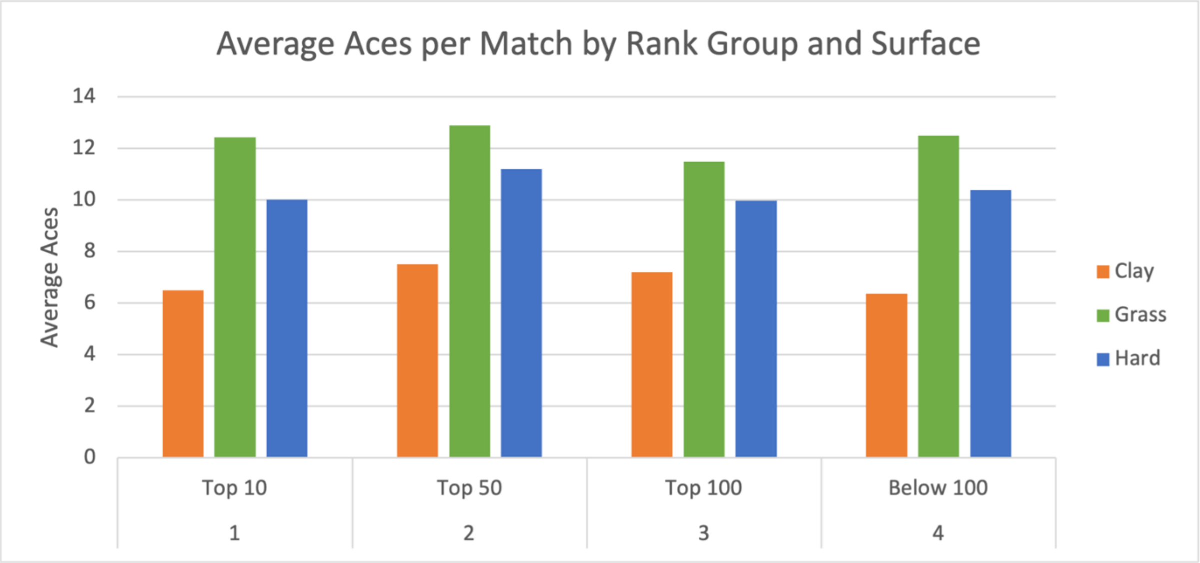

Fig. 1 – Aces per Match*. Top 50 players average more aces than Top 10*

- Across all surfaces, **Top 50** players hit the most aces — even more than the Top 10. This suggests they may rely more heavily on a strong serve to stay competitive, unlike Top 10 players who tend to have more all-round skills.
- As expected, **grass courts** produce the highest average aces for all groups, followed by **hard** and then **clay**, due to surface speed and bounce dynamics.

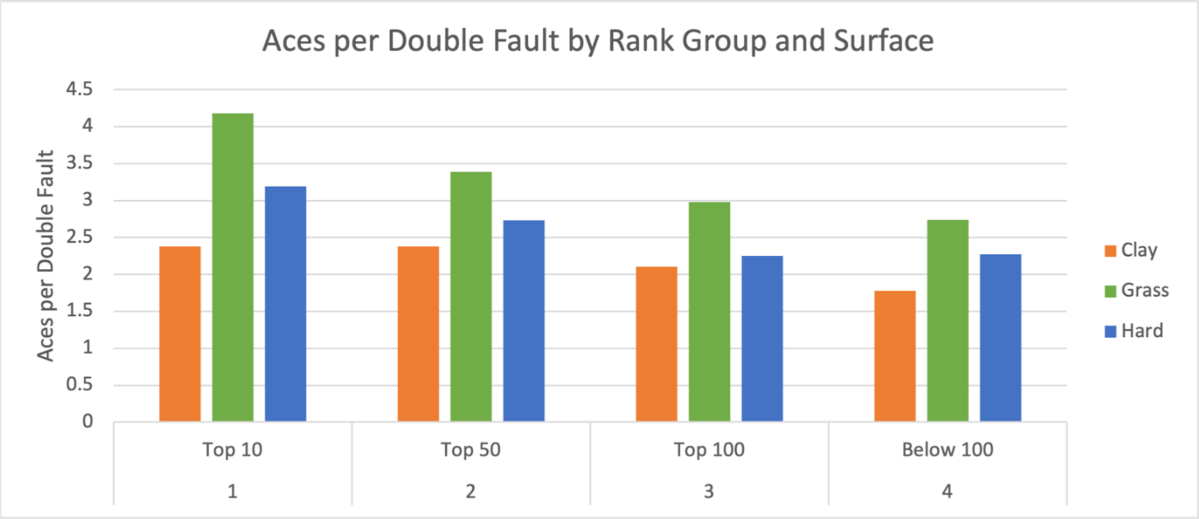

***Fig. 2 – Aces/Double Faults Ratio**. Top 10 players show the best serve efficiency.*

- **Top 10** players show the best aces/double faults ratio, reflecting both power and precision. **Below 100** players perform worst, hinting at a high-risk but less consistent serving style.
- This ratio helps **normalize serve efficiency** better than looking at aces or double faults alone.

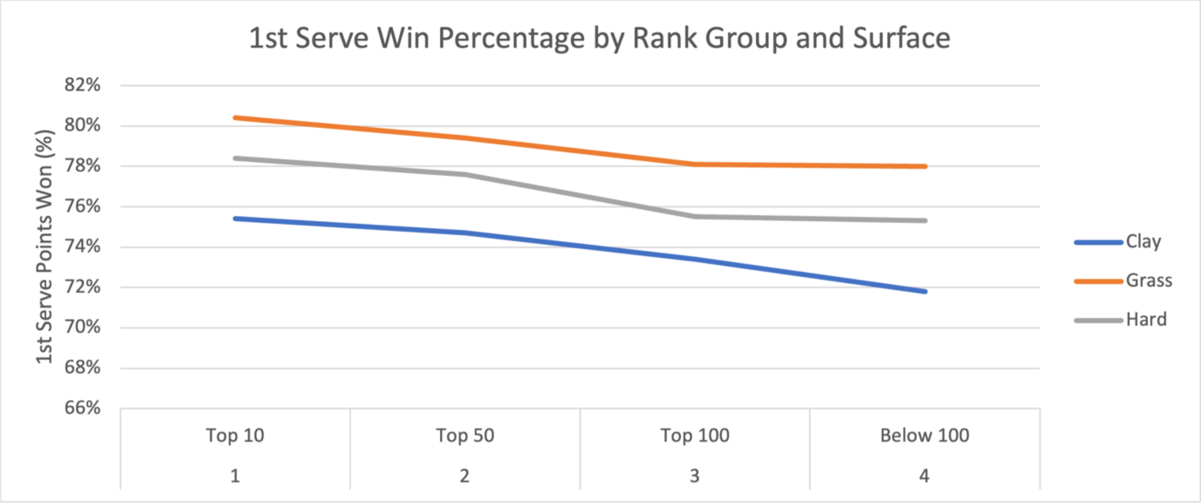

***Fig. 3 – First Serve Win %**. Higher-ranked players win more points on first serve.*

- A clear rank gradient appears: **Top 10 > Top 50 > Top 100 > Below 100**, showing how effectively higher-ranked players convert first serves into points.
- Surface trends hold as expected: **grass > hard > clay**, consistent with known differences in court speed.

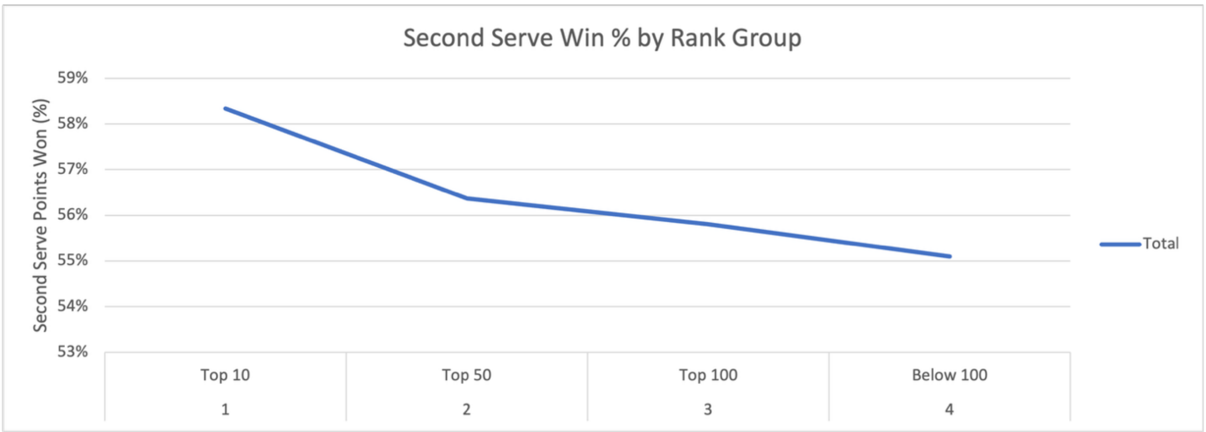

Fig. 4 – Second Serve Win %. *Top 10 players win significantly more second serve points.*

- The divide is striking: **Top 10 players win ~58%**, while **Below 100** players fall toward 55%. This suggests superior pressure handling and point construction at higher levels.
- Surface breakdown was excluded to focus on clearer rank-based patterns (surface effects were weaker and inconsistent).

🟦 Conclusion

This case study highlights how serve performance — especially **first and second serve win percentages** — clearly distinguishes the world's top players from the rest. By using SQL for large-scale data preparation and Excel for surface- and ranking-based visualizations, I identified clear trends and translated them into actionable insights.

From data cleaning and transformation in BigQuery to exploratory analysis and storytelling in Excel, this project demonstrates my ability to turn raw data into meaningful, decision-supporting insights. It's a small but powerful example of how data analysis can illuminate performance drivers — not only in tennis, but across competitive domains.

🟦 Limitations

- This analysis only includes *winner* statistics, which may introduce survivorship bias. Loser data could offer additional context.
- Rally length and net approach metrics were excluded due to missing data.
- No statistical significance testing was performed; future work could explore whether observed trends (e.g., serve win % differences) are statistically meaningful.
- Data is filtered to Grand Slam matches from 2000 onward for consistency — findings may not generalize to other tournaments or time periods.

🟦 Future Work

- **Include return game analysis** to better understand the full player profile, especially for top performers who excel in both serving and returning.
- **Incorporate loser data** to avoid survivorship bias and enable more balanced comparisons.
- **Explore temporal trends**, such as how serve performance has evolved from 2000 to 2022 across surfaces and rankings.
- **Add statistical significance testing** to validate observed trends between rank groups (e.g., t-tests on 1st serve win %).

📎 Appendix: SQL Queries Used

Below are the queries used to prepare the grouped dataset for analysis. The first query retrieves general metadata with rank groups; the second aggregates performance statistics per surface and rank group.

```
SELECT
  tourney_date,
  surface,
  tourney_name,
  round,
  winner_name,
  winner_rank,
  CASE
  WHEN SAFE_CAST(winner_rank AS INT64) <= 10 THEN 'Top 10'
  WHEN SAFE_CAST(winner_rank AS INT64) <= 50 THEN 'Top 50'
  WHEN SAFE_CAST(winner_rank AS INT64) <= 100 THEN 'Top 100'
  ELSE 'Below 100'
END AS winner_rank_group,

CASE
  WHEN SAFE_CAST(loser_rank AS INT64) <= 10 THEN 'Top 10'
  WHEN SAFE_CAST(loser_rank AS INT64) <= 50 THEN 'Top 50'
  WHEN SAFE_CAST(loser_rank AS INT64) <= 100 THEN 'Top 100'
  ELSE 'Below 100'
END AS loser_rank_group,
FROM `powerful-rhino-456217-i1.Tennis_data.ATP_matches_till_2022`
WHERE
  tourney_level = 'G'
  AND winner_rank IS NOT NULL
  AND loser_rank IS NOT NULL
  AND tourney_date >= 20000101
LIMIT 1000;


SELECT
  surface,
  CASE
    WHEN CAST(SAFE_CAST(winner_rank AS FLOAT64) AS INT64) <= 10 THEN 'Top 10'
    WHEN CAST(SAFE_CAST(winner_rank AS FLOAT64) AS INT64) <= 50 THEN 'Top 50'
    WHEN CAST(SAFE_CAST(winner_rank AS FLOAT64) AS INT64) <= 100 THEN 'Top 100'
    ELSE 'Below 100'
  END AS winner_rank_group,

  COUNT(*) AS matches_played,
  ROUND(AVG(SAFE_CAST(w_ace AS FLOAT64)), 2) AS avg_aces,
  ROUND(AVG(SAFE_CAST(w_df AS FLOAT64)), 2) AS avg_double_faults,

  ROUND(AVG(
    SAFE_CAST(w_1stIn AS FLOAT64) / NULLIF(SAFE_CAST(w_svpt AS FLOAT64), 0)
  ), 3) AS first_serve_pct,
  ROUND(AVG(
    SAFE_CAST(w_1stWon AS FLOAT64) / NULLIF(SAFE_CAST(w_1stIn AS FLOAT64), 0)
  ), 3) AS first_serve_win_pct,

  ROUND(AVG(
    SAFE_CAST(w_2ndWon AS FLOAT64) /
    NULLIF(SAFE_CAST(w_svpt AS FLOAT64) - SAFE_CAST(w_1stIn AS FLOAT64), 0)
  ), 3) AS second_serve_win_pct,

  ROUND(AVG(
    SAFE_CAST(w_bpSaved AS FLOAT64) / NULLIF(SAFE_CAST(w_bpFaced AS FLOAT64), 0)
  ), 3) AS bp_save_pct

FROM `powerful-rhino-456217-i1.Tennis_data.ATP_matches_till_2022`

WHERE
  tourney_level = 'G'
  AND tourney_date >= 20000101
  AND winner_rank IS NOT NULL
  AND w_svpt IS NOT NULL
  AND w_1stIn IS NOT NULL
  AND w_1stWon IS NOT NULL
  AND w_2ndWon IS NOT NULL
  AND w_ace IS NOT NULL
  AND w_df IS NOT NULL
  AND w_bpFaced IS NOT NULL
  AND w_bpSaved IS NOT NULL

GROUP BY
  surface, winner_rank_group
ORDER BY
  surface, winner_rank_group;
```

To keep the logic modular and maintainable, one could also use Common Table Expressions (CTEs) to filter and cast upfront. However, in this dataset, I found that casting inline gave more reliable results due to data type inconsistencies.