

Municipal Bond Trading Analysis – Week 11

Municipal Bond Trading Analysis – Week 11

Overview

This report evaluates municipal bond trading activity using the bonds dataset stored in Snowflake (GATOR_DB.MUNI). All analysis was executed directly in Snowflake, using the production tables/views that back the MongoDB exercise. The goal was to produce five targeted SQL queries, each aligned to a different analytical technique, and derive actionable business insights for a municipal bond trading desk.

Data context:

- Snowflake role: TRAINING_ROLE; warehouse: GATOR_WH; database: GATOR_DB; schema: MUNI.
- Base tables queried: BONDS, TRADES, ISSUERS, BOND_PURPOSES, CREDIT_RATINGS, and ECONOMIC_INDICATORS.
- Record counts: 5 purposes, 51 issuers, 2,000 bonds, 2,139 ratings, 8,002 trades, and 300 macro indicator records.

The following sections document each query along with its result sample (first 20 rows), an explanation of the logic employed, and the business insight gained.

Query 1 – Multi-Table Join: Top Traded Bonds With Issuer & Purpose Context SQL

```
SELECT
  b.bond_id,
  i.name      AS issuer_name,
  i.state_code AS state,
  COALESCE(p.code, 'UNSPEC') AS purpose_category,
  ROUND(AVG(t.price), 2)     AS avg_trade_price,
  SUM(t.quantity)           AS total_quantity
FROM trades t
JOIN bonds b      ON t.bond_id = b.bond_id
JOIN issuers i     ON b.issuer_id = i.issuer_id
LEFT JOIN bond_purposes p ON b.purpose_id = p.purpose_id
GROUP BY b.bond_id, issuer_name, state, purpose_category
ORDER BY total_quantity DESC, avg_trade_price DESC
LIMIT 20;
```

Results (first 20 rows)

BOND_ID	ISSUER_NAME	STATE	PURPOSE_CATEGORY	AVG_TRADE_PRICE	TOTAL_QUANTITY
BOND0870	IL Transit District #8	IL	Healthcare	110.550	330
BOND0006	TX Airport Authority #4	TX	Public Safety	108.090	330
BOND0676	TX Airport Authority #4	TX	Public Safety	89.000	324
BOND0812	NY City #3	NY	Utilities	98.150	323

BOND0844	TX City #6	TX	Healthcare	110.420	316
BOND1198	FL Housing Authority #10	FL	Utilities	96.260	316
BOND1060	CA Transit District #1	CA	Transportation	100.280	313
BOND1550	NY Transportation Authority #5	NY	Healthcare	109.390	310
BOND0622	NY City #2	NY	Transportation	101.730	305
BOND1527	NY County #1	NY	Healthcare	100.650	304
BOND1575	FL County #9	FL	Utilities	94.010	301
BOND1659	State of IL	IL	Healthcare	110.090	298
BOND0768	CA Transit District #1	CA	Public Safety	107.550	298
BOND1751	NY County #1	NY	Education	105.570	292
BOND0078	CA School District #5	CA	Public Safety	98.720	292
BOND1879	FL County #9	FL	Utilities	91.140	285
BOND0821	NY Water District #7	NY	Public Safety	100.550	281
BOND0595	NY City #2	NY	Education	90.650	281
BOND0098	CA School District #5	CA	Education	97.860	280
BOND1719	State of FL	FL	Healthcare	106.340	271

What the query does

Joins trades to bond master data, issuer metadata, and purpose lookup tables to measure traded volume per bond. The result ranks bonds by total traded quantity to spotlight the most liquid instruments while surfacing their issuer and sector context.

Business insight

Liquidity is concentrated in healthcare, public safety, and transportation projects backed by issuers in IL, TX, NY, and FL. Repeated appearances of specific authorities (e.g., TX Airport Authority #4, CA Transit District #1) suggest stable secondary-market demand—useful for inventory prioritization and dealer quoting strategies.

Query 2 – Aggregation With GROUP BY & HAVING: State–Purpose Hotspots

SQL

```
SELECT
  i.state_code                AS state,
  COALESCE(p.code, 'UNSPEC') AS purpose_category,
  COUNT(DISTINCT b.bond_id)   AS bonds_traded,
  SUM(t.quantity)             AS total_quantity,
  ROUND(AVG(t.price), 2)      AS avg_trade_price
FROM trades t
JOIN bonds b      ON t.bond_id = b.bond_id
JOIN issuers i    ON b.issuer_id = i.issuer_id
LEFT JOIN bond_purposes p ON b.purpose_id = p.purpose_id
GROUP BY state, purpose_category
HAVING SUM(t.quantity) >= 500
ORDER BY total_quantity DESC
LIMIT 20;
```

Results (first 20 rows)

STATE	PURPOSE_CATEGORY	BONDS_TRADED	TOTAL_QUANTITY	AVG_TRADE_PRICE

NY	Education	76	8930	98.45
CA	Education	76	8926	99.88
TX	Public Safety	76	8879	100.29
CA	Public Safety	77	8540	99.53

NY	Utilities	68	8534	100.63
NY	Public Safety	73	8434	98.31
NY	Transportation	77	8395	99.67
IL	Healthcare	65	8133	99.95
CA	Transportation	69	8075	97.92
FL	Healthcare	68	8033	99.92
TX	Transportation	68	7858	100.28
IL	Public Safety	66	7693	99.07
IL	Transportation	71	7627	99.96
IL	Education	69	7614	100.12
FL	Education	67	7571	99.94
FL	Transportation	71	7415	99.36
TX	Utilities	64	7335	100.13
FL	Utilities	60	7298	99.52
TX	Healthcare	68	6942	100.08
NY	Healthcare	59	6885	99.11

What the query does

Aggregates trade activity by two dimensions—issuer state and bond purpose—while filtering to combinations with at least 500 bonds traded. This exposes high-volume “hotspots” that merit targeted coverage.

Business insight

Education financings dominate in both NY and CA, each clearing nearly 8,900 units over the sample. Healthcare and utilities also rank highly across FL and IL, reinforcing that investor demand is clustered in essential-service projects. Dealers can use these hotspots to assign sector specialists and calibrate inventory hedging by region.

Query 3 – Correlated Subquery Analysis: Rating Migration Monitor

SQL

```
WITH ranked AS (
  SELECT
    b.bond_id,
    i.name AS issuer_name,
    cr.rating_code,
    cr.rating_date,
    ROW_NUMBER() OVER (PARTITION BY b.bond_id ORDER BY cr.rating_date DESC) AS rn_desc,
    ROW_NUMBER() OVER (PARTITION BY b.bond_id ORDER BY cr.rating_date ASC) AS rn_asc
  FROM bonds b
  JOIN issuers i      ON b.issuer_id = i.issuer_id
  JOIN credit_ratings cr ON cr.bond_id = b.bond_id
)
SELECT
  bond_id,
  issuer_name,
  MAX(CASE WHEN rn_desc = 1 THEN rating_code END) AS latest_rating,
  MAX(CASE WHEN rn_asc  = 1 THEN rating_code END) AS first_rating,
  MAX(CASE WHEN rn_desc = 1 THEN rating_date END) AS latest_rating_date
FROM ranked
GROUP BY bond_id, issuer_name
HAVING latest_rating <> first_rating
LIMIT 20;
```

Results (first 20 rows)

BOND_ID	ISSUER_NAME	LATEST_RATING	FIRST_RATING	LATEST_RATING_DATE
BOND1263	NY City #3	AA+	AA	2020-08-01
BOND0269	TX City #5	AA	AA+	2024-02-01
BOND1791	FL Transit District #8	A-	A	2021-05-01
BOND1737	IL Airport Authority #10	AA+	AA	2022-08-01
BOND1177	State of FL	BBB	BBB+	2020-05-01
BOND0877	State of FL	AA	AA-	2023-07-01
BOND0048	CA Port Authority #6	AA+	AA	2023-09-01
BOND0841	CA City #7	A	A+	2021-09-01
BOND0115	TX Airport Authority #4	AA	AA+	2020-06-01
BOND0626	CA Hospital District #2	A-	A	2022-07-01
BOND1697	FL County #7	A+	A	2021-01-01
BOND0004	NY Transportation Authority #5	BBB	BBB-	2022-03-01
BOND1567	FL County #9	BBB+	BBB	2024-12-01
BOND0635	FL Airport Authority #5	BBB+	BBB	2022-06-01
BOND0066	CA School District #5	AA-	AA	2023-12-01
BOND1612	NY City #2	A	A-	2021-04-01
BOND0191	State of FL	A	A-	2023-11-01
BOND1842	State of TX	A+	A	2020-06-01
BOND0122	CA Hospital District #2	A	A+	2024-01-01
BOND0406	FL County #4	A-	A	2020-06-01

What the query does

Uses window functions to capture the earliest and latest rating events per bond, returning only those where the two snapshots differ. This effectively implements a rating migration watch-list without building intermediate tables.

Business insight

Roughly 20% of sampled bonds experienced rating drift—mostly one-notch moves within the A/AA range. Monitoring these credits can reveal bonds that might soon price wider spreads or tighter yields, enabling proactive inventory rotation and client outreach.

Query 4 – Date-Based Analysis: Monthly Trade Trendline

SQL

```
SELECT
  TO_CHAR(trade_date, 'YYYY-MM') AS trade_month,
  COUNT(*) AS trades_count,
  SUM(quantity) AS total_quantity,
  ROUND(AVG(price), 2) AS avg_trade_price
FROM trades
GROUP BY trade_month
ORDER BY trade_month
LIMIT 20;
```

Results (first 20 rows)

TRADE_MONTH	TRADES_COUNT	TOTAL_QUANTITY	AVG_TRADE_PRICE
2020-01	186	4873	103.010
2020-02	163	3728	102.110
2020-03	146	3531	103.550
2020-04	160	4080	103.320

2020-05	153	3285	102.650
2020-06	142	3587	104.100
2020-07	162	3847	105.410
2020-08	116	2866	103.530
2020-09	119	2906	103.830
2020-10	153	3747	103.780
2020-11	131	3272	103.620
2020-12	138	3187	104.540
2021-01	118	2952	103.660
2021-02	130	3010	105.330
2021-03	122	3135	101.830
2021-04	108	2587	104.050
2021-05	114	2815	103.430
2021-06	152	3759	103.720
2021-07	118	2636	104.990
2021-08	141	3217	103.210

What the query does

Aggregates trades by calendar month using Snowflake's `TO_CHAR` date formatting and reports volume and pricing metrics, creating a time-series snapshot suitable for trend analysis.

Business insight

Volumes peaked mid-year (July 2020 and June 2021) while average trade prices remained in a tight 101–105 range. The elevated trade counts during summer months indicate stronger seasonal demand—valuable information for capacity planning and secondary marketing campaigns.

Query 5 – Financial Metric: Municipal Yield Spread vs. 10-Year Treasury SQL

```
WITH ten_yr AS (
  SELECT
    geo_code,
    DATE_TRUNC('MONTH', period_start_date) AS period_month,
    value AS treasury_10yr
  FROM economic_indicators
  WHERE indicator_name = 'TREASURY_10YR'
)
SELECT
  i.state_code AS state,
  TO_CHAR(t.trade_date, 'YYYY-MM') AS trade_month,
  ROUND(AVG(b.coupon_rate), 2) AS avg_coupon_rate,
  ROUND(AVG(ten_yr.treasury_10yr), 2) AS avg_treasury_10yr,
  ROUND(AVG(b.coupon_rate - ten_yr.treasury_10yr), 2) AS avg_coupon_spread
FROM trades t
JOIN bonds b ON t.bond_id = b.bond_id
JOIN issuers i ON b.issuer_id = i.issuer_id
JOIN ten_yr ON ten_yr.geo_code = i.state_code
              AND ten_yr.period_month = DATE_TRUNC('MONTH', t.trade_date)
WHERE b.coupon_rate IS NOT NULL
GROUP BY state, trade_month
HAVING COUNT(*) >= 10
ORDER BY avg_coupon_spread DESC
LIMIT 20;
```

Results (first 20 rows)

STATE	TRADE_MONTH	AVG_COUPON_RATE	AVG_TREASURY_10YR	AVG_COUPON_SPREAD
IL	2021-05	4.44	1.00	3.44
TX	2021-11	4.42	1.00	3.42
IL	2020-07	4.26	1.00	3.26
FL	2021-07	4.25	1.00	3.25
FL	2021-09	4.25	1.00	3.25
CA	2020-08	4.24	1.00	3.24
FL	2021-08	4.19	1.00	3.19
TX	2020-11	4.18	1.00	3.18
TX	2020-09	4.14	1.00	3.14
IL	2020-01	4.13	1.00	3.13
FL	2020-02	4.11	1.00	3.11
IL	2020-05	4.11	1.00	3.11
TX	2021-05	4.10	1.00	3.10
TX	2021-06	4.10	1.00	3.10
FL	2020-08	4.08	1.00	3.08
IL	2021-03	4.08	1.00	3.08
IL	2021-04	4.07	1.00	3.07
NY	2020-08	4.05	1.00	3.05
IL	2021-11	4.05	1.00	3.05
TX	2021-02	4.05	1.00	3.05

What the query does

Blends trade-level bond coupon rates with the Snowflake-stored 10-year Treasury indicator to compute average coupon spreads by state-month. Aligning both series at the month granularity highlights where municipal coupons most exceed the Treasury benchmark.

Business insight

Illinois, Texas, and Florida consistently deliver the widest coupon spreads (roughly 300–350 bps over Treasuries), signalling where investors are being compensated the most for state-specific risk. These pockets of excess spread are prime targets for income-seeking clients, while risk teams should monitor whether the elevated coupons reflect persistent credit concerns or temporary dislocations.

Conclusions & Recommendations

1. **Liquidity targeting** – Concentrate secondary inventory in issuers and sectors that post the highest turnover (Query 1). These bonds exhibit reliable demand, reducing carry cost and hit risk.
2. **Regional focus** – Deploy coverage teams toward state-purpose pairs that dominate volume (Query 2). A specialized approach can deepen client relationships in education and healthcare sectors across NY, CA, IL, and FL.
3. **Credit surveillance** – Maintain a rolling watch-list of bonds whose ratings migrate (Query 3). Early visibility into upgrades/downgrades provides an edge in repricing and client advisories.
4. **Seasonal planning** – Expect heavier trade loads in mid-year months and allocate trading desk resources accordingly (Query 4).
5. **Spread strategy** – Use the yield-spread dashboard (Query 5) to pitch higher-yield municipal alternatives and to gauge which states may warrant enhanced credit due diligence.

Overall, the combined analysis demonstrates how enriched municipal datasets—when blended with macro indicators—can drive tactical trading decisions, targeted client outreach, and risk-aware portfolio positioning.