Dynamic Credit Utilization Forecasting
-Predictive & Prescriptive Segmentation

Submitted by
Group 3

RFP SQL QUERY AND RESULTS

Under the Guidance of Industry Mentor
Abhishek Mukherjee

And Faculty Mentor
Sundar Raj Vijayanagar

```
********************************************************************************
```

Objective 1: Segment customers into meaningful groups

```
********************************************************************************

================================================================================
```

Q1) We created the cc_segments view — now let's inspect segment distribution and sample customers.

```
--------------------------------------------------------------------------------
Description:
Turns two interpretable metrics — UTILIZATION_RATIO (how much of their credit limit a
customer uses) and FULL_PAY_PCT (discipline to pay statements in full) — into simple,
explainable segments that everyone in the business can understand. We CROSS JOIN a tiny
single-row quantiles view (cc_quantiles from Q6) so thresholds
are computed once, then apply CASE rules to label each customer as 'Safe High Spenders',
'Moderate Users', 'Revolvers', or 'Over-Leveraged'. We intentionally select only the
columns required by downstream steps to keep the view skinny and Spark-friendly.
Why this matters:
Before we model/prescribe actions, we need to see how customers are split into explainable
groups. A quick distribution tells Product/Marketing whether the portfolio is skewed to
"Revolvers", "Moderate Users", etc. This also validates that quantile thresholds produced a
reasonable mix (not all in one bucket).
Business Impact:
Segments turn raw ratios into business personas (Safe High Spenders, Moderate Users,
Revolvers, Over-Leveraged). Distribution and samples validate thresholds and give
Marketing/Risk a common language to act on.
================================================================================
SELECT
  SEGMENT,
  COUNT(*)                          AS customers,
  ROUND(AVG(UTILIZATION_RATIO),3)   AS avg_util,
  ROUND(AVG(FULL_PAY_PCT),3)        AS avg_full_pay
FROM workspace.credit_card_project.cc_segments
GROUP BY SEGMENT
ORDER BY customers DESC;

SELECT
  CUST_ID, SEGMENT, ROUND(UTILIZATION_RATIO,3), FULL_PAY_PCT, CREDIT_LIMIT, BALANCE
FROM workspace.credit_card_project.cc_segments
ORDER BY SEGMENT, UTILIZATION_RATIO DESC
LIMIT 40;
```

1.1

SEGMENT	customers	avg_util	avg_full_pay
Moderate Users	4475	0.32	0.116
Revolvers	3147	0.645	0.009
Safe High Spenders	1328	0.015	0.625

1.2

CUST_ID	SEGMENT	UTILIZATION_RATIO	FULL_PAY_PCT	CREDIT_LIMIT	BALANCE
C11246	Moderate Users	0.718	0	7000	5022.775729
C16745	Moderate Users	0.717	0	6500	4661.423201
C18505	Moderate Users	0.717	0	6000	4302.771935
C13862	Moderate Users	0.717	0	1000	716.970968
C17663	Moderate Users	0.717	0	2000	1433.404387
C18472	Moderate Users	0.716	0	3000	2149.205166
C13693	Moderate Users	0.716	0.083333	4500	3221.717702
C12620	Moderate Users	0.716	0	7000	5010.879703
C13407	Moderate Users	0.715	0	1000	715.044288
C13246	Moderate Users	0.715	0.083333	3150	2251.631886
C16399	Moderate Users	0.714	0	1200	857.313072
C13429	Moderate Users	0.714	0	1400	999.528562
C16717	Moderate Users	0.714	0	7500	5354.53794
C14542	Moderate Users	0.714	0	10500	7496.171727
C15478	Moderate Users	0.713	0	1750	1248.193121
C13648	Moderate Users	0.713	0	5650	4025.646358
C16901	Moderate Users	0.712	0	5000	3560.977973
C15261	Moderate Users	0.712	0.090909	2500	1780.188356
C17393	Moderate Users	0.712	0.083333	4000	2847.472954
C11443	Moderate Users	0.712	0	3000	2135.365916
C17105	Moderate Users	0.712	0	5000	3558.14254
C10658	Moderate Users	0.712	0.090909	3000	2134.513436
C17802	Moderate Users	0.711	0	10000	7114.235621
C15124	Moderate Users	0.711	0	3000	2133.901379
C13313	Moderate Users	0.711	0	8500	6045.192228
C11276	Moderate Users	0.711	0	2200	1564.180333
C16610	Moderate Users	0.711	0	1000	710.811464
C14416	Moderate Users	0.711	0	8000	5686.466725
C18447	Moderate Users	0.711	0	5000	3553.326258
C10512	Moderate Users	0.711	0	6000	4263.008734
C17657	Moderate Users	0.71	0	1200	852.530387
C17889	Moderate Users	0.71	0	2500	1775.884148
C13876	Moderate Users	0.71	0	1800	1278.502197
C16732	Moderate Users	0.71	0.125	1200	851.880648
C14228	Moderate Users	0.71	0	9500	6743.313868
C18367	Moderate Users	0.71	0	3500	2483.739581
C15709	Moderate Users	0.71	0	6500	4612.147149
C17607	Moderate Users	0.709	0.083333	1000	709.475496
C18973	Moderate Users	0.709	0.285714	1200	851.249818
C16199	Moderate Users	0.709	0	3500	2481.696036

```
===================================================================================
Q2) We created the cohort_profitability view — now let's compare lifecycle value by tenure × segment.
-----------------------------------------------------------------------------------
Description:
Groups customers by tenure bands (0–12, 13–24, 25–36, 37+) and segment to estimate a
revenue proxy from purchases (1.5% interchange). Uses a CTE to compute bands once and then
aggregates. Only the necessary columns are scanned.
Why this matters:
Find the sweet spots in lifecycle. If new customers (0–12m) in a given segment have high
revenue proxy, we double down on onboarding offers. If older cohorts slow, we plan re-
engagement.
Business Impact:
Shows where lifecycle ROI is strongest. Guides investment in onboarding vs. re-engagement,
ensuring marketing money is spent where incremental lifetime value is highest.
===================================================================================
SELECT
  SEGMENT, tenure_band, customers, avg_purchases, revenue_proxy
FROM workspace.credit_card_project.cohort_profitability
ORDER BY SEGMENT, tenure_band;
```

SEGMENT	tenure_band	customers	avg_purchases	revenue_proxy
Moderate Users	0-12	4475	1387.2	93115.56
Revolvers	0-12	3147	543.33	25647.85
Safe High Spenders	0-12	1328	799.04	15916.84

```
==================================================================================
Q3) We created the income_util_repayment view — now let's evaluate utilization & repayment by income band.
-----------------------------------------------------------------------------------
Description:
Bins customers by ANNUAL_INCOME and summarizes utilization and full-payment behavior per
band. Early column trimming (CTE) reduces shuffle and memory. Helps detect aggressive
credit use in certain income tiers.
Why this matters:
Fair, data-based pricing. If certain income bands routinely carry high utilization AND low
full-pay ratios, we need education or tighter limits; if the opposite, we can responsibly
expand.
Business Impact:
Guides fair, effective policy: which income tiers can handle more credit vs. where
repayment discipline lags. Supports compliance, product fairness, and strategic pricing.
===================================================================================
```

```sql
SELECT
  income_band, customers, avg_utilization, avg_full_pay_pct
FROM workspace.credit_card_project.income_util_repayment
ORDER BY customers DESC;
```

income_band	customers	avg_utilization	avg_full_pay_pct
10L-20L	3233	0.382	0.158
20L+	3227	0.39	0.153
5L-10L	1531	0.398	0.152
<5L	959	0.395	0.142

```
===================================================================================
```

Q4) We created the cash_advance_hotspots view — now let's find regions & tiers with cash-advance dependence.

```
-----------------------------------------------------------------------------------
Description:
Finds geographies with high reliance on cash advances (a risk signal). We pre-select only
the fields required from demographics and features, join, then aggregate to locate hot-
spots.
Why this matters:
Cash advances are expensive for customers and risky for lenders. Identifying hotspots
enables safer short-term credit products, better advice, or extra controls — improving NPS
and reducing losses.
Business Impact:
Pinpoints riskier geographies where customers rely on costly cash advances. Enables
interventions like cheaper products, education, or proactive controls to reduce losses and
improve satisfaction.
===================================================================================
SELECT
  REGION, CITY_TIER, customers, avg_cash_adv_depend
FROM workspace.credit_card_project.cash_advance_hotspots
ORDER BY avg_cash_adv_depend DESC, customers DESC;
```

REGION	CITY_TIER	customers	avg_cash_adv_depend
East	Semi-Urban	467	1273.888
West	Urban	540	1033.397
Central	Urban	523	68.302
South	Rural	289	16.489
East	Rural	289	11.171
North	Semi-Urban	451	8.428
West	Semi-Urban	410	3.634
South	Urban	582	3.339
East	Urban	533	2.878
Central	Metro	518	2.485
North	Rural	268	2.328
North	Urban	518	2.098
Central	Rural	266	2.024
East	Metro	538	1.83
West	Metro	557	1.777

South	Metro	560	1.641
North	Metro	477	1.493
West	Rural	267	1.442
South	Semi-Urban	458	1.415
Central	Semi-Urban	439	1.318

```
*********************************************************************************
```

Objective 2: Forecast future credit utilization under stress

```
*********************************************************************************
=================================================================================
```

Q5) We created the stress_params view — now let's display the scenario knobs (interest hike, repayment drop).

```
---------------------------------------------------------------------------------
Description:
Creates a tiny TEMP VIEW with knobs for scenario analysis: interest_hike_bps (basis points)
and repayment_drop_pct. Keeping parameters centralized avoids re-coding logic and lets you
iterate multiple scenarios quickly (baseline, mild stress, severe stress) without touching
downstream SQL.
Why this matters:
Executives will ask "what happens if rates rise by X or repayment drops by Y?". Keeping
these in one table makes scenario runs traceable and auditable (no hidden constants in
code).
Business Impact:
Keeps stress-test assumptions transparent. Leadership can change numbers once here and all
downstream models update, speeding decision-making and ensuring governance.
=================================================================================
SELECT *
FROM workspace.credit_card_project.stress_params;
```

interest_hike_bps	repayment_drop_pct
200	0.15

```
=================================================================================
```

Q6) We created the cc_util_forecast view — now let's review predicted utilization & risk under stress.

```
---------------------------------------------------------------------------------
Description:
Projects each customer's behavior forward under the chosen stress scenario. We:
  • Build a skinny CTE from cc_segments (only required columns),
  • CROSS JOIN the tiny stress_params,
  • Compute baseline utilization (util_pred_base), stressed repayment (full_pay_stress),
    stressed balance (balance_stress), stressed utilization (util_pred_stress,
    capped),    and a simple interpretable risk_score on [0,1].
Keeping the projection narrow reduces shuffles and memory.
Why this matters:
```

```
This is the forward-looking engine. We quickly verify ranges and surface extremes so Risk
can triage. Sorting by highest risk highlights accounts that are most sensitive to macro
shocks (good early-warning list).
Business Impact:
Shifts analysis from descriptive to forward-looking. Identifies accounts most stretched in
adverse scenarios so Risk can proactively manage exposure and Finance can plan capital
buffers.
=================================================================================
SELECT
  ROUND(AVG(util_pred_base),3)   AS avg_base_util,
  ROUND(AVG(util_pred_stress),3) AS avg_stress_util,
  ROUND(AVG(risk_score),3)       AS avg_risk
FROM workspace.credit_card_project.cc_util_forecast;

SELECT
  CUST_ID, SEGMENT, ROUND(util_pred_base,3), ROUND(util_pred_stress,3),
ROUND(risk_score,3), CREDIT_LIMIT, BALANCE
FROM workspace.credit_card_project.cc_util_forecast
ORDER BY risk_score DESC, util_pred_stress DESC
LIMIT 30;
```

6.1

avg_base_util	avg_stress_util	avg_risk
0.474	0.57	0.545

6.2

CUST_ID	SEGMENT	util_pred_base	util_pred_stress	risk_score	CREDIT_LIMIT	BALANCE
C15349	Revolvers	null	5	1	null	18.400472
C17140	Revolvers	15.966	5	1	50	795.497557
C13566	Revolvers	2.417	2.558	1	150	348.813275
C10591	Revolvers	2.134	2.274	1	1700	3457.086184
C18225	Revolvers	1.819	1.953	1	1000	1718.885963
C13235	Revolvers	1.67	1.802	1	1200	1884.252524
C12477	Revolvers	1.654	1.785	1	1000	1553.505165
C16614	Revolvers	1.64	1.771	1	1000	1540.478248
C18093	Revolvers	1.481	1.609	1	1000	1381.080028
C12853	Revolvers	1.474	1.602	1	2000	2748.868773
C17913	Revolvers	1.423	1.549	1	1000	1322.64748
C14499	Revolvers	1.421	1.548	1	8000	10571.41107
C14401	Revolvers	1.399	1.525	1	1800	2339.037958
C15141	Revolvers	1.386	1.512	1	2500	3215.903805
C17221	Revolvers	1.382	1.508	1	1000	1281.911087
C11905	Revolvers	1.377	1.502	1	1000	1276.941373
C16720	Revolvers	1.372	1.497	1	1500	1907.779363
C18223	Revolvers	1.357	1.482	1	1000	1256.81927
C16970	Revolvers	1.351	1.476	1	1000	1251.26572
C10543	Revolvers	1.35	1.475	1	1500	1874.601636
C13380	Revolvers	1.337	1.462	1	1200	1484.693487
C11478	Revolvers	1.328	1.452	1	1000	1227.548135
C17095	Revolvers	1.305	1.429	1	1500	1807.764778

C17479	Revolvers	1.305	1.429	1	1000	1205.024103
C15122	Revolvers	1.295	1.418	1	3000	3583.515054
C18848	Revolvers	1.282	1.406	1	1000	1182.080141
C10693	Revolvers	1.282	1.406	1	1200	1418.254924
C16689	Revolvers	1.282	1.405	1	1500	1772.610312
C11351	Revolvers	1.272	1.396	0.998	1000	1172.344218
C16591	Revolvers	1.263	1.386	0.993	1000	1162.7795

```
==================================================================================
```

Q7) We created the risk_heatmap_region_tier view — now let's see where risk clusters geographically.

```
----------------------------------------------------------------------------------
Description:
Joins a narrow demographic slice (region, city tier) with a narrow forecast slice (risk,
stressed utilization), then aggregates by REGION × CITY_TIER. Designed to aggregate after
trimming to reduce memory.
Why this matters:
Shows geographic concentration of risk. Helps regional managers plan playbooks (education,
collections), and informs marketing/branch strategy.
Business Impact:
Helps allocate field resources and design regional playbooks. Identifies where extra
collections support, education, or pricing changes may be needed to contain risk.
==================================================================================
SELECT
  REGION, CITY_TIER, customers, avg_risk, avg_stressed_util
FROM workspace.credit_card_project.risk_heatmap_region_tier
 ORDER BY avg_risk DESC, customers DESC;
```

REGION	CITY_TIER	customers	avg_risk	avg_stressed_util
East	Rural	289	0.565	0.602
East	Semi-Urban	467	0.562	0.602
Central	Urban	523	0.558	0.59
West	Metro	557	0.555	0.592
South	Semi-Urban	458	0.554	0.58
South	Rural	289	0.552	0.581
East	Urban	533	0.549	0.58
North	Rural	268	0.549	0.576
North	Semi-Urban	451	0.546	0.577
Central	Metro	518	0.544	0.566
South	Urban	582	0.543	0.567
South	Metro	560	0.54	0.557
Central	Semi-Urban	439	0.54	0.561
Central	Rural	266	0.54	0.555
East	Metro	538	0.538	0.554
North	Urban	518	0.538	0.559
West	Rural	267	0.538	0.566
North	Metro	477	0.535	0.545
West	Urban	540	0.532	0.549
West	Semi-Urban	410	0.53	0.548

```
*********************************************************************************
```

Objective 3: Recommend personalized credit strategies

```
*********************************************************************************

=================================================================================
```

Q8) We created the cc_prescriptions view — now let's summarize recommended limit/APR actions.

```
---------------------------------------------------------------------------------
Description:
Maps predicted risk and utilization to tangible decisions Operations can execute: increase
limit (by 10–20%), reduce APR for the safest customers, or tighten policy for the riskiest.
Rules are simple and auditable.
Projection is kept skinny for downstream consumption.
Why this matters:
Confirms the policy engine is behaving: safest customers should mostly get upgrades/APR
cuts; riskiest should see reductions or no increases. This table is the bridge from
analytics to operations.
Business Impact:
Turns risk forecasts into explainable operational levers. Safe customers get upgrades and
APR cuts, risky ones see reductions. Improves portfolio health while keeping decisions
auditable for compliance.
=================================================================================
SELECT
  CASE
    WHEN recommended_limit_change_pct > 0 THEN 'Proposed Limit Upgrade'
    WHEN recommended_limit_change_pct < 0 THEN 'Proposed Limit Reduce'
    ELSE 'No Change'
  END                              AS action_type,
  COUNT(*)                         AS customers,
  ROUND(AVG(risk_score),3)         AS avg_risk
FROM workspace.credit_card_project.cc_prescriptions
GROUP BY action_type
ORDER BY customers DESC;

SELECT
  CUST_ID, SEGMENT, risk_score, ROUND(util_pred_base,3), ROUND(util_pred_stress,3),
  recommended_limit_change_pct, recommended_apr_bps
FROM workspace.credit_card_project.cc_prescriptions
ORDER BY util_pred_base DESC
LIMIT 20;
```

8.1

action_type	customers	avg_risk
No Change	5443	0.428
Proposed Limit Reduce	2798	0.829
Proposed Limit Upgrade	709	0.323

8.2

CUST_ID	SEGMENT	risk_score	util_pred_base	util_pred_stress	recommended_limit_change_pct	Recommended_apr_bps
C17140	Revolvers	1	15.966	5	-0.10	100
C13566	Revolvers	1	2.417	2.558	-0.10	100
C10591	Revolvers	1	2.134	2.274	-0.10	100
C18225	Revolvers	1	1.819	1.953	-0.10	100
C13235	Revolvers	1	1.67	1.802	-0.10	100
C12477	Revolvers	1	1.654	1.785	-0.10	100
C16614	Revolvers	1	1.64	1.771	-0.10	100
C18093	Revolvers	1	1.481	1.609	-0.10	100
C12853	Revolvers	1	1.474	1.602	-0.10	100
C17913	Revolvers	1	1.423	1.549	-0.10	100
C14499	Revolvers	1	1.421	1.548	-0.10	100
C14401	Revolvers	1	1.399	1.525	-0.10	100
C15141	Revolvers	1	1.386	1.512	-0.10	100
C17221	Revolvers	1	1.382	1.508	-0.10	100
C11905	Revolvers	1	1.377	1.502	-0.10	100
C16720	Revolvers	1	1.372	1.497	-0.10	100
C18223	Revolvers	1	1.357	1.482	-0.10	100
C16970	Revolvers	1	1.351	1.476	-0.10	100
C10543	Revolvers	1	1.35	1.475	-0.10	100
C13380	Revolvers	1	1.337	1.462	-0.10	100

```
===================================================================================
```

Q9) We created the cc_actions table — now let's count rows and preview the ops-ready list.

```
-----------------------------------------------------------------------------------
Description:
Materializes the prescriptive recommendations in a compact Delta table. This decouples
analytics from operational systems so dashboards/exports can use a stable object without
recomputing heavy logic.
Why this matters:
This is the object CRM/credit-ops will actually pull. We verify row counts and skim a few
top candidates to ensure downstream jobs have a clean, stable source.
Business Impact:
This is the stable Delta table CRM/Credit Ops consumes. By validating here, we ensure
campaigns execute on trusted, production-ready data without recomputing heavy analytics
logic.
===================================================================================
SELECT COUNT(*) AS total_actions
FROM workspace.credit_card_project.cc_actions;
```

```sql
SELECT
  CUST_ID, SEGMENT, ROUND(risk_score,3), ROUND(util_pred_base,3),
ROUND(util_pred_stress,3),
  recommended_limit_change_pct, recommended_apr_bps
FROM workspace.credit_card_project.cc_actions
ORDER BY recommended_limit_change_pct DESC, util_pred_base DESC
LIMIT 30;
```

Total Action – 8950

9.2

CUST_ID	SEGMENT	risk_score	util_pred_base	util_pred_stress	Recommended_limit_change_pct	Recommended_apr_bps
C17796	Moderate Users	0.21	0.295	0.331	0.20	-100
C15208	Moderate Users	0.201	0.277	0.313	0.20	-100
C10611	Moderate Users	0.201	0.277	0.313	0.20	-100
C13408	Moderate Users	0.23	0.263	0.307	0.20	-100
C18954	Moderate Users	0.234	0.239	0.286	0.20	-100
C18719	Moderate Users	0.23	0.239	0.285	0.20	-100
C10584	Moderate Users	0.233	0.237	0.284	0.20	-100
C14358	Moderate Users	0.243	0.232	0.282	0.20	-100
C18609	Moderate Users	0.178	0.231	0.265	0.20	-100
C13621	Moderate Users	0.177	0.229	0.264	0.20	-100
C15398	Moderate Users	0.201	0.228	0.269	0.20	-100
C11300	Moderate Users	0.174	0.223	0.257	0.20	-100
C11612	Moderate Users	0.197	0.222	0.262	0.20	-100
C16389	Moderate Users	0.245	0.22	0.272	0.20	-100
C10306	Moderate Users	0.172	0.219	0.253	0.20	-100
C10587	Moderate Users	0.219	0.217	0.263	0.20	-100
C11471	Moderate Users	0.171	0.217	0.251	0.20	-100
C10952	Moderate Users	0.171	0.217	0.251	0.20	-100
C15628	Moderate Users	0.218	0.215	0.26	0.20	-100
C10352	Moderate Users	0.224	0.209	0.257	0.20	-100
C16016	Moderate Users	0.23	0.207	0.256	0.20	-100
C18926	Moderate Users	0.229	0.206	0.256	0.20	-100
C14400	Moderate Users	0.162	0.2	0.234	0.20	-100
C18379	Moderate Users	0.333	0.464	0.511	0.10	0
C11669	Moderate Users	0.397	0.425	0.492	0.10	0
C10310	Moderate Users	0.379	0.419	0.482	0.10	0
C15344	Moderate Users	0.381	0.387	0.454	0.10	0
C19104	Moderate Users	0.372	0.386	0.451	0.10	0
C17574	Moderate Users	0.398	0.378	0.45	0.10	0
C17006	Moderate Users	0.251	0.375	0.413	0.10	0

```
================================================================================
Q11) We created the cc_business_impact view — now let's rank segments by revenue uplift minus risk cost.
--------------------------------------------------------------------------------
Description:
Computes per-customer uplift and risk cost in a CTE (joining actions with limit) and then aggregates by segment. Outputs counts, average risk, incremental revenue, expected risk cost, and net impact. The design minimizes shuffles by trimming columns before JOIN/GROUP BY.
Why this matters:
Turns model outputs into money. This view tells leadership which segments will likely produce positive net impact after accounting for stress-loss cost, so budgets and targets can be set with confidence.
Business Impact:
Links model outputs to money. Leadership can prioritize investments in segments with the best net ROI while de-prioritizing or tightening risky ones. Data-driven budget allocation tool.
================================================================================
SELECT
  SEGMENT, customers, avg_risk, incr_spend, incr_revenue_est, risk_cost_est, net_impact_est
FROM workspace.credit_card_project.cc_business_impact
ORDER BY net_impact_est DESC;

SELECT
  SUM(customers)                                AS customers,
  ROUND(AVG(avg_risk),3)                        AS avg_risk_weighted,
  ROUND(SUM(incr_revenue_est),2)                AS total_incr_revenue_est,
  ROUND(SUM(risk_cost_est),2)                   AS total_risk_cost_est,
  ROUND(SUM(net_impact_est),2)                  AS total_net_impact_est
FROM workspace.credit_card_project.cc_business_impact;
```

11.1

SEGMENT	customers	avg_risk	incr_spend	incr_revenue_est	risk_cost_est	net_impact_est
Safe High Spenders	1328	0.195	9182.5	137.737	0	137.737
Moderate Users	4475	0.524	50155.5	752.332	184896.787	-184144.4552
Revolvers	3147	0.723	14652.5	219.787	243113.868	-242894.0809

11.2

customers	avg_risk_weighted	total_incr_revenue_est	total_risk_cost_est	total_net_impact_est
8950	0.481	1109.86	428010.66	-426900.8

```
================================================================================
Q12) We created the target_limit_upgrades view — now let's pull the shortlist of safe customers for outreach.
--------------------------------------------------------------------------------
Description:
Creates an ops-ready shortlist of the safest customers who are recommended for limit
upgrades and are most likely to respond (higher baseline utilization). We JOIN to a narrow
demographic slice for context and order by util_pred_base to prioritize the highest
potential first.
Why this matters:
Directly powers outreach. These are low-risk, high-propensity customers prioritized by
current utilization. You can filter by city/age/income to create sub-campaigns.
Business Impact:
Directly empowers CRM/Sales campaigns. A ranked, low-risk call list increases conversion
efficiency and boosts spend without materially adding risk.
================================================================================
SELECT
  CUST_ID, SEGMENT, ROUND(risk_score,3), ROUND(util_pred_base,3),
  recommended_limit_change_pct,
  AGE, ANNUAL_INCOME, CITY_TIER, EDUCATION
FROM workspace.credit_card_project.target_limit_upgrades
ORDER BY util_pred_base DESC
LIMIT 200;
```

CUST_ID	SEGMENT	Risk _score	Util _pred _base	Reco mmended _limit_ change_pct	AGE	ANNUAL _INCOME	CITY_TIER	EDUCATION
C18379	Moderate Users	0.333	0.464	0.10	24	1625749	Semi-Urban	High School
C11669	Moderate Users	0.397	0.425	0.10	68	1647872	Urban	Graduate
C10310	Moderate Users	0.379	0.419	0.10	60	1965122	Urban	Postgraduate
C15344	Moderate Users	0.381	0.387	0.10	69	207212	Urban	Graduate
C19104	Moderate Users	0.372	0.386	0.10	62	531686	Rural	Postgraduate
C17574	Moderate Users	0.398	0.378	0.10	36	2335260	Semi-Urban	Graduate
C17006	Moderate Users	0.251	0.375	0.10	50	2952456	Urban	Postgraduate
C16816	Moderate Users	0.298	0.364	0.10	29	2800538	Rural	Graduate
C15235	Moderate Users	0.291	0.339	0.10	57	252050	Rural	Professional
C10375	Moderate Users	0.391	0.319	0.10	49	1501876	Urban	Postgraduate
C10885	Moderate Users	0.391	0.318	0.10	40	2328259	Semi-Urban	Doctorate
C18534	Moderate Users	0.316	0.314	0.10	55	2871736	Urban	Professional
C18785	Moderate Users	0.266	0.31	0.10	28	451991	Metro	Doctorate
C18886	Moderate Users	0.386	0.31	0.10	46	855914	Semi-Urban	Graduate
C11394	Moderate Users	0.398	0.301	0.10	41	1569597	Metro	Doctorate
C15294	Moderate Users	0.387	0.301	0.10	57	2466687	Rural	Postgraduate
C12292	Moderate Users	0.384	0.296	0.10	61	232611	Rural	Graduate
C17796	Moderate Users	0.21	0.295	0.20	40	2216803	Semi-Urban	Postgraduate

C11541	Moderate Users	0.378	0.294	0.10	54	1519946	Urban	Professional
C13802	Moderate Users	0.281	0.278	0.10	50	898638	Semi-Urban	Postgraduate
C18782	Moderate Users	0.317	0.277	0.10	45	868581	Semi-Urban	Postgraduate
C15208	Moderate Users	0.201	0.277	0.20	57	2653280	Metro	Doctorate
C10611	Moderate Users	0.201	0.277	0.20	61	2903456	Semi-Urban	High School
C10751	Moderate Users	0.393	0.276	0.10	44	275012	Rural	Postgraduate
C14048	Moderate Users	0.272	0.275	0.10	30	2405182	Urban	Doctorate
C18579	Moderate Users	0.364	0.272	0.10	65	2562121	Rural	Postgraduate
C12336	Moderate Users	0.343	0.272	0.10	66	815171	Rural	Postgraduate
C10805	Moderate Users	0.391	0.271	0.10	21	2915475	Metro	High School
C15661	Moderate Users	0.391	0.27	0.10	45	2802939	Urban	High School
C13408	Moderate Users	0.23	0.263	0.20	26	1963063	Semi-Urban	Postgraduate
C17835	Moderate Users	0.324	0.26	0.10	29	2198917	Metro	Graduate
C18428	Moderate Users	0.373	0.26	0.10	49	504583	Semi-Urban	Graduate
C12531	Moderate Users	0.384	0.258	0.10	58	697611	Urban	Graduate
C17825	Moderate Users	0.335	0.255	0.10	57	1928885	Semi-Urban	High School
C10040	Moderate Users	0.383	0.255	0.10	27	973511	Urban	High School
C10736	Moderate Users	0.382	0.253	0.10	34	2276257	Rural	Postgraduate
C12213	Moderate Users	0.399	0.252	0.10	25	2241239	Rural	Graduate
C14447	Moderate Users	0.26	0.251	0.10	42	1356472	Metro	Postgraduate
C12846	Moderate Users	0.39	0.25	0.10	58	2011071	Metro	Graduate
C10855	Moderate Users	0.345	0.25	0.10	35	1279193	Urban	Graduate
C16658	Moderate Users	0.331	0.248	0.10	50	2729224	Urban	Postgraduate
C16983	Moderate Users	0.309	0.245	0.10	36	1828235	Urban	Postgraduate
C12474	Moderate Users	0.378	0.245	0.10	35	2544465	Metro	High School
C17521	Moderate Users	0.376	0.243	0.10	27	2957528	Semi-Urban	Postgraduate
C18797	Moderate Users	0.328	0.242	0.10	68	1730270	Metro	Graduate
C12897	Moderate Users	0.328	0.241	0.10	40	258014	Semi-Urban	Postgraduate
C18954	Moderate Users	0.234	0.239	0.20	63	1447708	Metro	Professional
C18719	Moderate Users	0.23	0.239	0.20	55	1199191	Semi-Urban	Professional
C14083	Moderate Users	0.326	0.239	0.10	42	1914249	Urban	Graduate
C10584	Moderate Users	0.233	0.237	0.20	56	1595748	Urban	Postgraduate
C18578	Moderate Users	0.397	0.236	0.10	61	359067	Metro	High School
C15164	Moderate Users	0.349	0.236	0.10	32	1178264	Urban	Graduate
C14358	Moderate Users	0.243	0.232	0.20	66	2813602	Metro	Postgraduate
C10901	Moderate Users	0.385	0.232	0.10	36	765642	Metro	Postgraduate
C13954	Moderate Users	0.388	0.231	0.10	44	2511914	Metro	Graduate
C18394	Moderate Users	0.298	0.231	0.10	37	2207648	Metro	Graduate
C14324	Moderate Users	0.274	0.231	0.10	51	512249	Urban	Graduate
C18609	Moderate Users	0.178	0.231	0.20	40	2987079	Rural	Graduate
C13401	Moderate Users	0.394	0.23	0.10	54	1111060	Semi-Urban	Postgraduate
C12032	Moderate Users	0.321	0.229	0.10	29	357952	Semi-Urban	Graduate
C13621	Moderate Users	0.177	0.229	0.20	59	2109558	Urban	Postgraduate
C16298	Moderate Users	0.393	0.229	0.10	29	1834389	Urban	Postgraduate
C18568	Moderate Users	0.369	0.228	0.10	23	1057107	Rural	Graduate
C15398	Moderate Users	0.201	0.228	0.20	45	2261782	Urban	Graduate
C11300	Moderate Users	0.174	0.223	0.20	46	521539	Semi-Urban	High School
C11612	Moderate Users	0.197	0.222	0.20	60	1078091	Urban	Graduate
C14765	Moderate Users	0.297	0.221	0.10	22	1863623	Metro	High School
C16389	Moderate Users	0.245	0.22	0.20	63	671207	Urban	Postgraduate
C11344	Moderate Users	0.365	0.22	0.10	49	464541	Semi-Urban	Graduate
C10306	Moderate Users	0.172	0.219	0.20	69	701907	Metro	Graduate
C13326	Moderate Users	0.388	0.219	0.10	29	1118210	Semi-Urban	Postgraduate
C18125	Moderate Users	0.292	0.219	0.10	49	213683	Urban	Professional
C16069	Moderate Users	0.374	0.218	0.10	64	510385	Metro	Graduate
C10587	Moderate Users	0.219	0.217	0.20	56	2111707	Urban	Graduate
C11471	Moderate Users	0.171	0.217	0.20	51	2276020	Rural	Professional
C10952	Moderate Users	0.171	0.217	0.20	67	1447265	Urban	Graduate
C18480	Moderate Users	0.387	0.216	0.10	22	1759164	Urban	High School
C15628	Moderate Users	0.218	0.215	0.20	24	2306741	Urban	Graduate
C17685	Moderate Users	0.376	0.215	0.10	24	1914756	Urban	Doctorate
C15825	Moderate Users	0.353	0.213	0.10	36	2012699	Urban	Graduate
C12915	Moderate Users	0.291	0.211	0.10	41	2881007	Urban	Postgraduate
C18026	Moderate Users	0.398	0.21	0.10	32	206246	Urban	Professional

C12987	Moderate Users	0.384	0.21	0.10	68	1912911	Urban	Postgraduate
C10352	Moderate Users	0.224	0.209	0.20	50	2628376	Metro	Doctorate
C16132	Moderate Users	0.397	0.208	0.10	32	2939164	Rural	High School
C16016	Moderate Users	0.23	0.207	0.20	55	2675077	Rural	High School
C18926	Moderate Users	0.229	0.206	0.20	69	854617	Metro	Professional
C15134	Moderate Users	0.396	0.206	0.10	48	2174555	Urban	Graduate
C12157	Moderate Users	0.309	0.205	0.10	25	866866	Metro	Professional
C10739	Moderate Users	0.321	0.203	0.10	21	1571401	Metro	High School
C10238	Moderate Users	0.348	0.203	0.10	24	600387	Urban	Graduate
C13398	Moderate Users	0.394	0.202	0.10	53	2225506	Urban	Postgraduate
C18707	Moderate Users	0.355	0.201	0.10	59	2317300	Metro	Graduate
C15636	Moderate Users	0.336	0.201	0.10	58	1744226	Rural	Postgraduate
C15218	Moderate Users	0.399	0.201	0.10	49	298170	Metro	Graduate
C12295	Moderate Users	0.379	0.201	0.10	54	705382	Metro	Postgraduate
C14400	Moderate Users	0.162	0.2	0.20	49	2422679	Semi-Urban	Professional
C14898	Moderate Users	0.258	0.199	0.10	52	2959801	Metro	High School
C12538	Moderate Users	0.187	0.198	0.10	22	2671797	Metro	Doctorate
C12588	Moderate Users	0.377	0.197	0.10	53	639486	Urban	Graduate
C15567	Moderate Users	0.376	0.195	0.10	68	2586253	Semi-Urban	Postgraduate
C15223	Moderate Users	0.352	0.195	0.10	26	926604	Metro	Postgraduate
C19147	Moderate Users	0.333	0.194	0.10	45	398079	Semi-Urban	Postgraduate
C12890	Moderate Users	0.375	0.192	0.10	29	717922	Semi-Urban	Graduate
C16034	Moderate Users	0.383	0.192	0.10	52	2145956	Urban	Doctorate
C10806	Moderate Users	0.394	0.191	0.10	41	2052371	Metro	Postgraduate
C16299	Moderate Users	0.398	0.19	0.10	62	2594472	Semi-Urban	Graduate
C10289	Moderate Users	0.302	0.19	0.10	68	412192	Metro	Postgraduate
C11505	Moderate Users	0.205	0.19	0.10	25	222642	Semi-Urban	Professional
C13195	Moderate Users	0.181	0.19	0.10	61	2551562	Rural	Professional
C16565	Moderate Users	0.156	0.189	0.10	63	2833647	Semi-Urban	Postgraduate
C17917	Moderate Users	0.359	0.189	0.10	58	1385126	Metro	High School
C19004	Moderate Users	0.301	0.189	0.10	27	2091053	Semi-Urban	High School
C19027	Moderate Users	0.317	0.189	0.10	45	2332228	Metro	Graduate
C11657	Moderate Users	0.156	0.188	0.10	21	1701404	Rural	Graduate
C15667	Moderate Users	0.392	0.188	0.10	59	2267830	Semi-Urban	Graduate
C18333	Moderate Users	0.372	0.187	0.10	64	318297	Metro	Professional
C18330	Moderate Users	0.396	0.187	0.10	58	2595713	Semi-Urban	Graduate
C11237	Moderate Users	0.184	0.187	0.10	63	2619435	Metro	Graduate
C11611	Moderate Users	0.371	0.186	0.10	63	2450403	Urban	High School
C17466	Moderate Users	0.347	0.185	0.10	67	1084499	Urban	Professional
C11765	Moderate Users	0.319	0.185	0.10	33	282195	Urban	Postgraduate
C18796	Moderate Users	0.269	0.184	0.10	49	1958496	Urban	High School
C14204	Moderate Users	0.25	0.184	0.10	63	2765147	Semi-Urban	Graduate
C15994	Moderate Users	0.394	0.183	0.10	54	2794917	Semi-Urban	Graduate
C15229	Moderate Users	0.356	0.183	0.10	23	2282810	Metro	Postgraduate
C17262	Moderate Users	0.298	0.183	0.10	27	1416201	Metro	Postgraduate
C14774	Moderate Users	0.297	0.182	0.10	61	1613140	Metro	Postgraduate
C10670	Moderate Users	0.369	0.182	0.10	29	1519631	Semi-Urban	High School
C12019	Moderate Users	0.318	0.181	0.10	58	696632	Urban	Postgraduate
C15428	Moderate Users	0.369	0.181	0.10	67	539012	Urban	Professional
C14667	Moderate Users	0.176	0.181	0.10	38	973255	Urban	Postgraduate
C13479	Moderate Users	0.393	0.181	0.10	60	1825207	Urban	Graduate
C15680	Moderate Users	0.321	0.181	0.10	21	788370	Rural	Postgraduate
C15479	Moderate Users	0.224	0.181	0.10	65	235224	Metro	Graduate
C18293	Moderate Users	0.345	0.181	0.10	64	681881	Semi-Urban	Graduate
C18898	Moderate Users	0.2	0.18	0.10	36	2130410	Urban	Graduate
C15358	Moderate Users	0.248	0.18	0.10	52	2588202	Semi-Urban	Graduate
C18822	Moderate Users	0.296	0.18	0.10	34	2227464	Urban	Doctorate
C18586	Moderate Users	0.256	0.179	0.10	36	2813827	Rural	Postgraduate
C18804	Moderate Users	0.344	0.178	0.10	49	1259831	Metro	Graduate
C14996	Moderate Users	0.151	0.178	0.10	45	2369208	Metro	Graduate
C10378	Moderate Users	0.368	0.178	0.10	51	412348	Semi-Urban	Graduate
C14407	Moderate Users	0.392	0.178	0.10	54	2895584	Metro	Graduate
C10514	Moderate Users	0.271	0.177	0.10	46	1057584	Urban	Graduate
C16902	Moderate Users	0.324	0.177	0.10	45	1699594	Urban	Postgraduate

C11177	Moderate Users	0.367	0.176	0.10	57	2957873	Urban	Postgraduate
C14536	Moderate Users	0.367	0.176	0.10	48	2473878	Urban	High School
C17886	Moderate Users	0.15	0.176	0.10	31	628341	Semi-Urban	Professional
C10720	Moderate Users	0.342	0.176	0.10	52	2661207	Urban	Graduate
C13997	Moderate Users	0.342	0.176	0.10	31	480030	Semi-Urban	Postgraduate
C16448	Moderate Users	0.15	0.176	0.10	45	1781678	Metro	Postgraduate
C19096	Moderate Users	0.366	0.175	0.10	64	1262361	Urban	Postgraduate
C10034	Moderate Users	0.351	0.175	0.10	36	1390971	Urban	Professional
C18387	Moderate Users	0.374	0.175	0.10	38	652356	Rural	Postgraduate
C10477	Moderate Users	0.293	0.174	0.10	32	1530131	Metro	Graduate
C10975	Moderate Users	0.341	0.173	0.10	55	1504581	Urban	Postgraduate
C12176	Moderate Users	0.293	0.173	0.10	61	2102458	Urban	Graduate
C18808	Moderate Users	0.341	0.173	0.10	65	1021350	Semi-Urban	Graduate
C18786	Moderate Users	0.365	0.173	0.10	53	203740	Metro	High School
C10295	Moderate Users	0.385	0.173	0.10	61	2388407	Metro	Professional
C15530	Moderate Users	0.317	0.172	0.10	31	2377941	Semi-Urban	Graduate
C12521	Moderate Users	0.384	0.172	0.10	24	2951387	Semi-Urban	Professional
C18328	Moderate Users	0.292	0.171	0.10	45	307083	Urban	Postgraduate
C14637	Moderate Users	0.388	0.17	0.10	26	1619909	Semi-Urban	Graduate
C18409	Moderate Users	0.307	0.17	0.10	21	1503050	Urban	Graduate
C13096	Moderate Users	0.339	0.17	0.10	65	1189233	Urban	Postgraduate
C19025	Moderate Users	0.363	0.169	0.10	49	2733596	Rural	Doctorate
C11143	Moderate Users	0.363	0.169	0.10	53	1187965	Urban	Graduate
C14124	Moderate Users	0.363	0.169	0.10	30	1231145	Urban	Professional
C10075	Moderate Users	0.363	0.169	0.10	30	1158281	Metro	Postgraduate
C12658	Moderate Users	0.363	0.169	0.10	21	2592457	Metro	Professional
C15003	Moderate Users	0.339	0.169	0.10	52	2585648	Urban	Graduate
C15607	Moderate Users	0.377	0.168	0.10	27	2713159	Metro	Professional
C11935	Moderate Users	0.362	0.168	0.10	28	1361969	Semi-Urban	Graduate
C12236	Moderate Users	0.356	0.168	0.10	62	913138	Urban	Graduate
C15562	Moderate Users	0.306	0.167	0.10	60	2799992	Urban	Graduate
C19168	Moderate Users	0.362	0.167	0.10	39	2582343	Semi-Urban	Graduate
C13180	Moderate Users	0.313	0.166	0.10	34	1174117	Metro	Professional
C16197	Moderate Users	0.144	0.165	0.10	62	2717774	Urban	Doctorate
C10170	Moderate Users	0.288	0.165	0.10	52	995657	Semi-Urban	Graduate
C17025	Moderate Users	0.392	0.164	0.10	68	1948986	Urban	Professional
C14594	Moderate Users	0.312	0.164	0.10	68	226920	Semi-Urban	Graduate
C18904	Moderate Users	0.384	0.163	0.10	31	2411513	Semi-Urban	Postgraduate
C16758	Moderate Users	0.207	0.163	0.10	38	2583403	Metro	Professional
C11226	Moderate Users	0.349	0.162	0.10	66	2025595	Urban	Professional
C13959	Moderate Users	0.287	0.161	0.10	55	2912126	Rural	Postgraduate
C11076	Moderate Users	0.359	0.161	0.10	63	2852201	Rural	Professional
C17701	Moderate Users	0.359	0.16	0.10	45	1870974	Urban	Graduate
C10529	Moderate Users	0.142	0.16	0.10	30	1302953	Urban	Graduate
C17181	Moderate Users	0.262	0.16	0.10	45	2828685	Metro	Postgraduate
C15393	Moderate Users	0.334	0.16	0.10	24	1179680	Metro	Graduate
C18474	Moderate Users	0.382	0.16	0.10	64	2844295	Metro	Postgraduate
C11430	Moderate Users	0.382	0.159	0.10	37	759811	Metro	Graduate
C16459	Moderate Users	0.141	0.159	0.10	65	1218829	Metro	Graduate
C15338	Moderate Users	0.325	0.159	0.10	55	1191010	Semi-Urban	High School
C11797	Moderate Users	0.31	0.159	0.10	47	1408531	Urban	Graduate
C19058	Moderate Users	0.348	0.159	0.10	57	1067422	Rural	Professional
C18951	Moderate Users	0.272	0.159	0.10	25	1194556	Metro	High School
C16155	Moderate Users	0.285	0.158	0.10	45	1844380	Metro	Graduate

```
================================================================================
```

Q13) We created the apr_ladder_summary view — now let's check concentration of APR changes by income × segment.

```
--------------------------------------------------------------------------------
Description:
Combines prescriptive APR recommendations with income bands to show where pricing changes
concentrate.Subqueries keep only necessary columns, then we GROUP BY to summarize counts
and average APR deltas.
Why this matters:
Supports compliance and product fairness reviews. Ensures adjustments are explainable and
not clustered in a way that looks arbitrary or biased; helps tune the APR ladder for growth
vs. risk.
Business Impact:
Demonstrates APR changes are explainable and balanced across segments and income bands.
Supports fairness in pricing strategy and builds trust with regulators/compliance.
================================================================================
SELECT
  SEGMENT, income_band, customers, apr_cuts, apr_hikes, avg_apr_bps
FROM workspace.credit_card_project.apr_ladder_summary
ORDER BY SEGMENT, income_band;
```

SEGMENT	Income_band	customers	apr_cuts	apr_hikes	avg_apr_bps
Moderate Users	10L-20L	1635	119	560	27
Moderate Users	20L+	1637	140	548	24.9
Moderate Users	5L-10L	746	51	264	28.6
Moderate Users	<5L	457	24	151	27.8
Revolvers	10L-20L	1083	0	778	71.8
Revolvers	20L+	1141	0	815	71.4
Revolvers	5L-10L	552	0	385	69.7
Revolvers	<5L	371	0	261	70.4
Safe High Spenders	10L-20L	515	338	0	-65.6
Safe High Spenders	20L+	449	285	0	-63.5
Safe High Spenders	5L-10L	233	157	0	-67.4
Safe High Spenders	<5L	131	81	0	-61.8

```
================================================================================
```

Q14) We created the upgrade_candidates view — now let's size the likely-to-respond pool by segment.

```
---------------------------------------------------------------------------------
Description:
Builds a compact view of customers with low modeled risk and mid-to-high baseline
utilization who also have a recommended positive limit change. Uses a subquery to pre-trim
columns, then groups by segment to size the opportunity and report average
risk/utilization.
Why this matters:
Gives planning a fast read on "how big is the opportunity?" and "what's the average
risk/utilization in it?" so they can set campaign volume, budget, and guardrails.
Business Impact:
Directs campaign planning. Quantifies pool size and average profile of safe-but-constrained
customers who are most likely to generate incremental revenue from upgrades.
=================================================================================
SELECT
  SEGMENT, candidates, avg_current_util, avg_risk
FROM workspace.credit_card_project.upgrade_candidates
ORDER BY candidates DESC;
```

SEGMENT	candidates	avg_current_util	avg_risk
Moderate Users	97	0.255	0.319