Marco 2 do Projeto: Classificação em Base ao Risco Relativo

Maior Risco de Crédito:

```
CREATE OR REPLACE TABLE risco-relativo.credito.mau AS
WITH dados AS (
  SELECT
    age,
    more_90_days_overdue,
    using_lines_not_secured_personal_assets,
    total_loan,
    --clean_loan_type,
    number_dependents_median,
    last_month_salary_median,
    default_flag,
    NTILE(4) OVER (ORDER BY age) AS quartil_idade,
    NTILE(4) OVER (ORDER BY more_90_days_overdue) AS quartil_days,
    NTILE(4) OVER (ORDER BY using_lines_not_secured_personal_assets) AS
quartil_ativo,
    NTILE(4) OVER (ORDER BY total_loan) AS quartil_emprestimos,
    --CASE WHEN clean_loan_type = 'Real Estate' THEN 1 ELSE 0 END AS tipo_credito,
    NTILE(4) OVER (ORDER BY number_dependents_median) AS quartil_dependente,
    NTILE(4) OVER (ORDER BY last_month_salary_median) AS quartil_salario
  FROM
    `risco-relativo.credito.full_join`
),

riscos AS (
  SELECT
    'Idade' AS variavel,
    quartil_idade AS quartil,,
    AVG(CAST(default_flag AS FLOAT64)) AS incidencia,
    AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join` ) AS risco_relativo,
    ROW_NUMBER() OVER (PARTITION BY 'Idade' ORDER BY AVG(CAST(default_flag AS
FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64)) FROM
`risco-relativo.credito.full_join`) DESC) AS rn,
    AVG(age) AS valor
  FROM
    dados
  GROUP BY
    quartil
  UNION ALL
  SELECT
    'Dias de Atraso' AS variavel,
    quartil_days AS quartil,
```

```
    COUNT(more_90_days_overdue) AS quantidade,
    AVG(CAST(default_flag AS FLOAT64)) AS incidencia,
    AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`) AS risco_relativo,
    ROW_NUMBER() OVER (PARTITION BY 'Dias de Atraso' ORDER BY AVG(CAST(default_flag
AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64)) FROM
`risco-relativo.credito.full_join`) DESC) AS rn,
    AVG(more_90_days_overdue) AS valor
  FROM
    dados
  GROUP BY
    quartil
  UNION ALL
  SELECT
    'Uso Limite de Credito' AS variavel,
    quartil_ativo AS quartil,
    AVG(CAST(default_flag AS FLOAT64)) AS incidencia,
    AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`) AS risco_relativo,
    ROW_NUMBER() OVER (PARTITION BY 'Uso Limite de Credito' ORDER BY
AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`) DESC) AS rn,
    AVG(using_lines_not_secured_personal_assets) AS valor
  FROM
    dados
  GROUP BY
    quartil
  UNION ALL
  SELECT
    'Total de Empréstimos' AS variavel,
    quartil_emprestimos AS quartil,
    AVG(CAST(default_flag AS FLOAT64)) AS incidencia,
    AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`) AS risco_relativo,
    ROW_NUMBER() OVER (PARTITION BY 'Total de Empréstimos' ORDER BY
AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`)DESC) AS rn,
    AVG(total_loan) AS valor
  FROM
    dados
  GROUP BY
    quartil
  UNION ALL
  SELECT
```

```
    'Dependente' AS variavel,
    quartil_dependente AS quartil,
    AVG(CAST(default_flag AS FLOAT64)) AS incidencia,
    AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`) AS risco_relativo,
    ROW_NUMBER() OVER (PARTITION BY 'Total de Empréstimos' ORDER BY
AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`)DESC) AS rn,
    AVG(number_dependents_median) AS valor
  FROM
    dados
  GROUP BY
    quartil
  UNION ALL
  SELECT
    'Salario' AS variavel,
    quartil_salario AS quartil,
    AVG(CAST(default_flag AS FLOAT64)) AS incidencia,
    AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`) AS risco_relativo,
    ROW_NUMBER() OVER (PARTITION BY 'Salario' ORDER BY AVG(CAST(default_flag AS
FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64)) FROM
`risco-relativo.credito.full_join`) DESC) AS rn,
    AVG(last_month_salary_median) AS valor
  FROM
    dados
  GROUP BY
    quartil
),

maiores_riscos AS (
  SELECT
    variavel,
    quartil,
    valor,
    incidencia,
    risco_relativo
  FROM
    riscos
  WHERE
    rn = 1
)

SELECT * FROM maiores_riscos ORDER BY risco_relativo DESC;
```

Menor Risco de Crédito:

```
CREATE OR REPLACE TABLE risco-relativo.credito.bons AS
WITH dados AS (
  SELECT
    age,
    more_90_days_overdue,
    using_lines_not_secured_personal_assets,
    total_loan,
    --clean_loan_type,
    number_dependents_median,
    last_month_salary_median,
    default_flag,
    NTILE(4) OVER (ORDER BY age) AS quartil_idade,
    NTILE(4) OVER (ORDER BY more_90_days_overdue) AS quartil_days,
    NTILE(4) OVER (ORDER BY using_lines_not_secured_personal_assets) AS
quartil_ativo,
    NTILE(4) OVER (ORDER BY total_loan) AS quartil_emprestimos,
    --CASE WHEN clean_loan_type = 'Real Estate' THEN 1 ELSE 0 END AS tipo_credito,
    NTILE(4) OVER (ORDER BY number_dependents_median) AS quartil_dependente,
    NTILE(4) OVER (ORDER BY last_month_salary_median) AS quartil_salario
  FROM
    `risco-relativo.credito.full_join`
),

riscos AS (
  SELECT
    'Idade' AS variavel,
    quartil_idade AS quartil,
    AVG(CAST(default_flag AS FLOAT64)) AS incidencia,
    AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join` ) AS risco_relativo,
    ROW_NUMBER() OVER (PARTITION BY 'Idade' ORDER BY AVG(CAST(default_flag AS
FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64)) FROM
`risco-relativo.credito.full_join`) DESC) AS rn,
    AVG(age) AS valor
  FROM
    dados
  GROUP BY
    quartil
  UNION ALL
  SELECT
    'Dias de Atraso' AS variavel,
    quartil_days AS quartil,
    AVG(CAST(default_flag AS FLOAT64)) AS incidencia,
```

```
    AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`) AS risco_relativo,
    ROW_NUMBER() OVER (PARTITION BY 'Dias de Atraso' ORDER BY AVG(CAST(default_flag
AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64)) FROM
`risco-relativo.credito.full_join`) DESC) AS rn,
    AVG(more_90_days_overdue) AS valor
  FROM
    dados
  GROUP BY
    quartil
  UNION ALL
  SELECT
    'Uso Limite de Credito' AS variavel,
    quartil_ativo AS quartil,
    AVG(CAST(default_flag AS FLOAT64)) AS incidencia,
    AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`) AS risco_relativo,
    ROW_NUMBER() OVER (PARTITION BY 'Uso Limite de Credito' ORDER BY
AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`) DESC) AS rn,
    AVG(using_lines_not_secured_personal_assets) AS valor
  FROM
    dados
  GROUP BY
    quartil
  UNION ALL
  SELECT
    'Total de Empréstimos' AS variavel,
    quartil_emprestimos AS quartil,
    AVG(CAST(default_flag AS FLOAT64)) AS incidencia,
    AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`) AS risco_relativo,
    ROW_NUMBER() OVER (PARTITION BY 'Total de Empréstimos' ORDER BY
AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`)DESC) AS rn,
    AVG(total_loan) AS valor
  FROM
    dados
  GROUP BY
    quartil
  UNION ALL
  SELECT
    'Dependente' AS variavel,
    quartil_dependente AS quartil,
```

```
    AVG(CAST(default_flag AS FLOAT64)) AS incidencia,
    AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`) AS risco_relativo,
    ROW_NUMBER() OVER (PARTITION BY 'Total de Empréstimos' ORDER BY
AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`)DESC) AS rn,
    AVG(number_dependents_median) AS valor
  FROM
    dados
  GROUP BY
    quartil
  UNION ALL
  SELECT
    'Salario' AS variavel,
    quartil_salario AS quartil,
    AVG(CAST(default_flag AS FLOAT64)) AS incidencia,
    AVG(CAST(default_flag AS FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64))
FROM `risco-relativo.credito.full_join`) AS risco_relativo,
    ROW_NUMBER() OVER (PARTITION BY 'Salario' ORDER BY AVG(CAST(default_flag AS
FLOAT64)) / (SELECT AVG(CAST(default_flag AS FLOAT64)) FROM
`risco-relativo.credito.full_join`) DESC) AS rn,
    AVG(last_month_salary_median) AS valor
  FROM
    dados
  GROUP BY
    quartil
),

menores_riscos AS (
  SELECT
    variavel,
    quartil,
    valor,
    incidencia,
    risco_relativo
  FROM
    riscos
  WHERE
    rn = 4
)

SELECT * FROM menores_riscos oRDER BY risco_relativo DESC;
```

Query Consultas:

```
—segmentaçao em base a risco relativo variavel DYMMY
CREATE OR REPLACE TABLE `risco-relativo.credito.dummy` AS
WITH dados AS (
  SELECT
    user_id,
    age,
    more_90_days_overdue,
    number_times_delayed_payment_loan_30_59_days,
    number_times_delayed_payment_loan_60_89_days,
    total_loan,
    last_month_salary_median,
    number_dependents_median,
    using_lines_not_secured_personal_assets,
    NTILE(4) OVER (ORDER BY age) AS quartil_idade,
    NTILE(4) OVER (ORDER BY more_90_days_overdue) AS quartil_90days,
    NTILE(4) OVER (ORDER BY number_times_delayed_payment_loan_30_59_days) AS
quartil_30days,
    NTILE(4) OVER (ORDER BY number_times_delayed_payment_loan_60_89_days) AS
quartil_60days,
    NTILE(4) OVER (ORDER BY total_loan) AS quartil_empretismo,
    NTILE(4) OVER (ORDER BY last_month_salary_median) AS quartil_salario,
    NTILE(4) OVER (ORDER BY number_dependents_median) AS quartil_dependente,
    NTILE(4) OVER (ORDER BY using_lines_not_secured_personal_assets) AS
quartil_ativo
  FROM
    `risco-relativo.credito.full_join`
),

riscos AS (
  SELECT
    user_id,
    quartil_idade,
    more_90_days_overdue,
    number_times_delayed_payment_loan_30_59_days,
    number_times_delayed_payment_loan_60_89_days,
    quartil_empretismo,
    quartil_salario,
    quartil_dependente,
    quartil_ativo,
    CASE WHEN quartil_idade = 1 THEN 1 ELSE 0 END AS idade_risco,
    --CASE WHEN more_90_days_overdue > 1 THEN 1 ELSE 0 END AS dias90_risco,
    --CASE WHEN number_times_delayed_payment_loan_30_59_days > 1 THEN 1 ELSE 0 END
AS dias30_risco,
```

```sql
    --CASE WHEN number_times_delayed_payment_loan_60_89_days > 1 THEN 1 ELSE 0 END
AS dias60_risco,
    CASE WHEN quartil_90days = 4 THEN 1 ELSE 0 END AS dias90_risco,
    CASE WHEN quartil_30days = 4 THEN 1 ELSE 0 END AS dias30_risco,
    CASE WHEN quartil_60days = 4 THEN 1 ELSE 0 END AS dias60_risco,
    CASE WHEN quartil_empretismo = 1 THEN 1 ELSE 0 END AS empretismo_risco,
    CASE WHEN quartil_salario = 1 THEN 1 ELSE 0 END AS salario_risco,
    CASE WHEN quartil_dependente = 4 THEN 1 ELSE 0 END AS dependente_risco,
    CASE WHEN quartil_ativo = 4 THEN 1 ELSE 0 END AS uso_limite_risco
  FROM
    dados
),

pontuacao AS (
  SELECT
    user_id,
    idade_risco + dias90_risco + dias30_risco + dias60_risco + empretismo_risco +
salario_risco + dependente_risco + uso_limite_risco AS pontuacao
  FROM
    riscos
),

classificacao AS (
  SELECT
    user_id,
    CASE WHEN pontuacao.pontuacao >= 3 THEN 1 ELSE 0 END AS classificacao
  FROM
    pontuacao
)

SELECT
  r.user_id,
  r.idade_risco,
  r.dias90_risco,
  r.dias60_risco,
  r.dias30_risco,
  r.empretismo_risco,
  r.salario_risco,
  r.dependente_risco,
  r.uso_limite_risco,
  p.pontuacao,
  c.classificacao
FROM
  riscos r
```

```sql
JOIN
  pontuacao p
ON
  r.user_id = p.user_id
JOIN
  classificacao c
ON
  r.user_id = c.user_id;

--identificar
SELECT COUNT(classificacao)
from risco-relativo.credito.pontuacao_classificacao
WHERE classificacao = 1
```

Matriz de Confusão:

```
--Avaliação e Ajuste: Após classificar os indivíduos com base no risco
relativo(bons, mau), você pode avaliar o desempenho do modelo usando uma matriz de
confusão, por exemplo, para verificar a precisão das classificações e ajustar o
limiar, se necessário, para melhorar o desempenho do modelo.

CREATE OR REPLACE TABLE `risco-relativo.credito.matriz` AS
WITH metrics AS (
  SELECT
    COUNTIF(classificacao = 1 AND classificacao = 1) AS true_positive,
    COUNTIF(classificacao = 1 AND classificacao = 0) AS false_positive,
    COUNTIF(classificacao = 0 AND classificacao = 1) AS false_negative,
    COUNTIF(classificacao = 0 AND classificacao = 0) AS true_negative
  FROM
    `risco-relativo.credito.dummy`
)

SELECT
  true_positive,
  false_positive,
  false_negative,
  true_negative,
  (true_positive + true_negative) / (true_positive + false_positive +
false_negative + true_negative) AS accuracy,
  true_positive / (true_positive + false_positive) AS precision,
  true_positive / (true_positive + false_negative) AS recall,
  2 * ( (true_positive / (true_positive + false_positive)) * (true_positive /
(true_positive + false_negative)) ) /
  ( (true_positive / (true_positive + false_positive)) + (true_positive /
(true_positive + false_negative)) ) AS f1_score
FROM
  metrics
```