



BestMarket

Etude sur les retours clients réalisés / questions et réponses

Thierry Monjo

15/05/2024

1) Contexte et expression du besoin



- *Un besoin urgent et impérieux du service client d'interroger la satisfaction client afin d'ajuster sa stratégie*
- *Des données collectées concernant les retours clients :*
 - *à partir de sources diverses (téléphone, email, réseaux sociaux)*
 - *pour des services divers (service après-vente, livraison, expérience en magasin, drive).*
 - *concernant plusieurs typologies de produits (alimentaire, loisirs, high-tech, maison).*
- *Une liste de questions concernant la satisfaction client et d'indicateurs clés de performance à évaluer*

2) Sauvegarde et stockage de la BDD

- *L'outil utilisé est MySQL Workbench 8.0*
- *A partir d'une requête SQL communiquée, création de la base de données **customer_data_feedback** avec les tables **retour_client** et **produit***
- *Réalisation d'une **sauvegarde** intégrale par la fonction Export de l'outil (dump)*
- *Bonnes pratiques :*
 - *une copie de cette sauvegarde sur un **second support** conservé dans un **lieu différent***
 - *l'accès à la base de données ou à ses sauvegardes doit être **contrôlé et sécurisé***

3) Méthodologie suivie

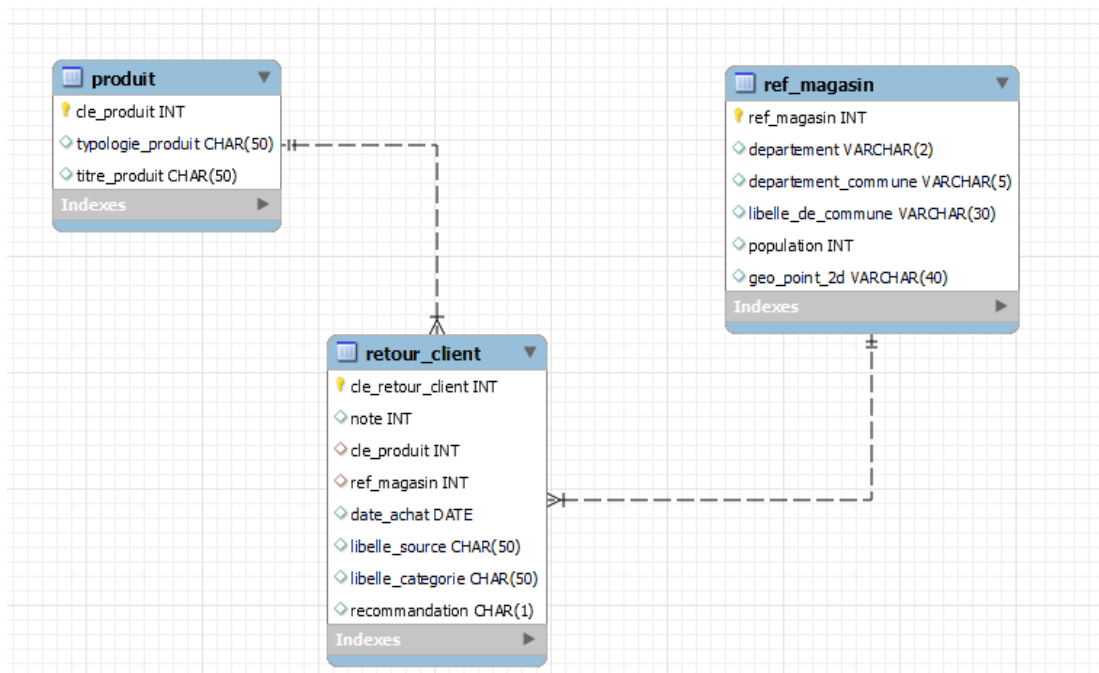
Dictionnaire de données

La base de données créée est enrichie de la table **ref_magasin**

Nom du champs	Type de données	Taille	Contrainte	Description
cle_retour_client	INT		Clé primaire	ID unique pour les retours clients
note	INT			Note donnée par le client, comprise entre 0 et 10, la note est la réponse à la question : "Sur une échelle de 0 à 10 quelle est la probabilité que vous recommandiez notre entreprise à votre entourage ?"
cle_produit	INT			ID des produits
ref_magasin	INT			ID des magasins
date_achat	DATE			Date à laquelle l'achat du client a eu lieu
libelle_source	CHAR	50		Libellé de la source d'où provient le retour client (Réseaux sociaux, téléphone, email)
libelle_categorie	CHAR	50		Libellé de la catégorie du retour client (Drive, service après-vente, qualité produit, expérience en magasin, livraison)
recommandation	CHAR			Recommandation laissée par le client à la question 'Recommandez vous l'entreprise?' True / False
cle_produit	INT		Clé primaire	ID unique pour les produits
titre_produit	CHAR	50		Libellé des produits
typologie_produit	INT			Typologie des produits (Alimentaire, High-tech etc...)
ref_magasin	INT		Clé primaire	ID unique pour les magasins
departement	CHAR	2		Département d'implantation du magasin
departement_commune	CHAR	5		Code département commune
libelle_de_commune	CHAR	30		Commune d'implantation du magasin
population	INT			Population de la commune concernée
geo_point_2d	CHAR	40		Coordonnées GPS du magasin

3) Méthodologie suivie

Visualisation du schéma relationnel dans MySQL Workbench 8.0



- Visualisation originelle du schéma relationnel (reverse engineering)
- Création de la table **ref_magasin** et mise à jour du schéma relationnel

3) Méthodologie suivie



Création et chargement des données de la table ref_magasin

- *Création physique de la table **ref_magasin** dans la base de données (forward engineering)*
- *Import du fichier csv des données de la table*
- *La base de données **customer_data_feedback** est complète et pleinement fonctionnelle*
- *Sauvegarde par export de la base de données réalisée par sécurité*

4) Requêtes SQL et Analyses

```
1  -- Quel est le nombre de retours clients sur la livraison ?
2  • SELECT
3      COUNT(*) AS 'nb_retour_clients_livraison'
4  FROM
5      retour_client
6  WHERE
7      libelle_categorie = 'livraison';
```

	nb_retour_clients_livraison
▶	639

```
1  -- Quelle est la liste des notes des clients sur les réseaux sociaux sur les TV ?
2  • SELECT DISTINCT
3      note AS 'liste_notes'
4  FROM
5      retour_client
6      JOIN
7      produit USING (cle_produit)
8  WHERE
9      libelle_source = 'réseaux sociaux'
10     AND titre_produit = 'TV';
```

	liste_notes
▶	10
	9
	8

Que des notes ≥ 8 / des clients satisfaits

4) Requêtes SQL et Analyses

```
1  -- Quelle est la note moyenne pour chaque catégorie de produit ?
2  -- (classé de la meilleure à la moins bonne)
3  • SELECT
4      typologie_produit, ROUND(AVG(note), 2) AS 'note_moyenne'
5  FROM
6      retour_client
7      JOIN
8      produit USING (cle_produit)
9  GROUP BY typologie_produit
10 ORDER BY AVG(note) DESC;
```

	typologie_produit	note_moyenne
►	High-Tech	8.16
	Loisirs	8.09
	Alimentaire	8.04
	Maison	7.85

Des notes moyennes favorables, dans un mouchoir de poche

```
1  -- Quels sont les 5 magasins avec les meilleures notes moyennes ?
2  • SELECT
3      ref_magasin,
4      departement,
5      libelle_de_commune,
6      ROUND(AVG(note), 2) AS 'note_moyenne'
7  FROM
8      retour_client
9      JOIN
10     ref_magasin USING (ref_magasin)
11  GROUP BY ref_magasin
12  ORDER BY note_moyenne DESC
13  LIMIT 5;
```

	ref_magasin	departement	libelle_de_commune	note_moyenne
►	75	75	Paris 14e Arrondissement	8.73
	78	91	Saint-Pierre-du-Perray	8.55
	62	75	Paris 19e Arrondissement	8.50
	23	75	Paris 11e Arrondissement	8.48
	19	77	Coulommiers	8.45

4) Requêtes SQL et Analyses

```
1  -- Quels sont les magasins qui ont plus de 12 feedbacks sur le drive ?
2  • SELECT
3      ref_magasin,
4      departement,
5      libelle_de_commune,
6      COUNT(*) AS 'nb_retours'
7  FROM
8      retour_client
9      JOIN
10     ref_magasin USING (ref_magasin)
11  WHERE
12     libelle_categorie = 'drive'
13  GROUP BY ref_magasin
14  HAVING nb_retours > 12
15  ORDER BY nb_retours DESC;
```

	ref_magasin	departement	libelle_de_commune	nb_retours
▶	67	95	Éragny	14
	63	94	Ivry-sur-Seine	13
	45	75	Paris 12e Arrondissement	13

```
1  -- Quel est le classement des départements par note ?
2  • SELECT
3      departement, ROUND(AVG(note), 2) AS 'note_moyenne'
4  FROM
5      retour_client
6      JOIN
7      ref_magasin USING (ref_magasin)
8  GROUP BY departement
9  ORDER BY note_moyenne DESC;
```

	departement	note_moyenne
▶	95	8.14
	75	8.11
	94	8.06
	91	8.05
	77	8.04
	92	8.03
	78	8.02
	93	7.94

Val d'Oise (95) et Paris dans le peloton de tête

4) Requêtes SQL et Analyses

```
1  -- Quelle est la typologie de produit qui apporte
2  -- le meilleur service après-vente ?
3  • SELECT
4      typologie_produit,
5      ROUND(AVG(note), 2) AS 'moyenne_note'
6  FROM
7      retour_client
8      JOIN
9      produit USING (cle_produit)
10 WHERE
11     libelle_categorie = 'service après-vente'
12 GROUP BY typologie_produit
13 ORDER BY moyenne_note DESC;
```

	typologie_produit	moyenne_note
►	Loisirs	8.51
	High-Tech	8.12
	Alimentaire	8.03
	Maison	7.88

```
1  -- Quelle est la note moyenne sur l'ensemble des boissons ?
2  • SELECT
3      ROUND(AVG(note), 2) AS 'note_moyenne_boissons'
4  FROM
5      retour_client
6      JOIN
7      produit USING (cle_produit)
8  WHERE
9      titre_produit LIKE 'Boissons%';
```

	note_moyenne_boissons
►	8.32

4) Requêtes SQL et Analyses

```
1  -- Quel est le classement des jours de la semaine où l'expérience client
2  -- est la meilleure expérience en magasin ?
3  • SELECT
4      DAYNAME(date_achat) AS 'jour_semaine',
5      ROUND(AVG(note), 2) AS 'note_moyenne_magasin'
6  FROM
7      retour_client
8  WHERE
9      libelle_categorie = 'expérience en magasin'
10 GROUP BY jour_semaine
11 ORDER BY note_moyenne_magasin DESC;
```

	jour_semaine	note_moyenne_magasin
►	Saturday	8.34
	Sunday	8.18
	Friday	8.07
	Thursday	8.04
	Wednesday	7.99
	Tuesday	7.95
	Monday	7.74

Les meilleures notes pour le week-end

```
1  -- Sur quel mois a-t-on le plus de retour sur le service après-vente ?
2  • SELECT
3      MONTHNAME(date_achat) AS 'mois', COUNT(*) AS 'nb_retours'
4  FROM
5      retour_client
6  WHERE
7      libelle_categorie = 'service après-vente'
8  GROUP BY mois
9  ORDER BY nb_retours DESC;
```

	mois	nb_retours
►	October	55
	September	53
	June	53
	August	52
	March	52
	November	52
	January	52
	May	52
	April	52
	July	48
	February	44
	December	38

L'automne, le mois de l'équipement ?

4) Requêtes SQL et Analyses

```
1  -- -- Quel est le pourcentage de recommandations client ?
2  •  SELECT DISTINCT
3      recommandation,
4      COUNT(*) AS nb,
5      ROUND((COUNT(*) / (SELECT
6                          COUNT(*)
7                          FROM
8                          retour_client)) * 100,
9              2) AS pourcentage
10 FROM
11     retour_client
12 GROUP BY recommandation;
```

	recommandation	nb	pourcentage
▶	1	2115	70.50
		674	22.47
	0	211	7.03

70,50 % ont fait une recommandation

4) Requêtes SQL et Analyses

```
1  -- Quels sont les magasins qui ont une note inférieure à la moyenne ?
2  • SELECT
3      ref_magasin,
4      departement,
5      libelle_de_commune,
6      ROUND(AVG(note), 2) AS 'note_moyenne'
7  FROM
8      retour_client
9      JOIN
10         ref_magasin USING (ref_magasin)
11  GROUP BY ref_magasin
12  HAVING note_moyenne < (SELECT
13      AVG(note)
14  FROM
15      retour_client
16  WHERE
17      libelle_categorie = 'expérience en magasin')
18  ORDER BY note_moyenne;
```

ref_magasin	departement	libelle_de_commune	note_mc
60	78	Buchelay	7.38
81	92	Nanterre	7.44
82	91	Montgeron	7.53
46	75	Paris 15e Arrondissement	7.56
55	93	Rosny-sous-Bois	7.59
80	77	Lognes	7.62
24	92	Levallois-Perret	7.62
8	93	Aubervilliers	7.66
44	94	Fontenay-sous-Bois	7.67
74	77	Chelles	7.70
57	95	Herblay	7.73
76	78	Carrières-sur-Seine	7.74
68	91	Fleury-Mérogis	7.79
66	91	Les Ulis	7.82
18	91	Saint-Germain-lès-Arpajon	7.83
25	78	Conflans-Sainte-Honorine	7.83
54	75	Paris 2e Arrondissement	7.84
51	93	Saint-Denis	7.84
65	78	Saint-Cyr-l'École	7.85
63	94	Ivry-sur-Seine	7.86
20	77	Provins	7.87
34	92	Asnières-sur-Seine	7.89
33	75	Paris 18e Arrondissement	7.89
58	78	Coignières	7.90
14	75	Paris 17e Arrondissement	7.90
6	95	Osny	7.90
13	91	Ballainvilliers	7.92
45	75	Paris 12e Arrondissement	7.93
64	77	Ozoir-la-Ferrière	7.93
79	93	Livry-Gargan	7.95
53	92	Puteaux	7.97
38	92	Rueil-Malmaison	7.97
73	91	Viry-Châtillon	7.97
7	75	Paris 20e Arrondissement	8.00
59	75	Paris 1er Arrondissement	8.00
47	78	Freneuse	8.03

4) Requêtes SQL et Analyses

```
1  -- Quelles sont les typologies produits qui ont amélioré leur moyenne entre le 1er et le 2ème trimestre 2021 ?
2  • WITH T2 AS (
3      SELECT
4          typologie_produit,
5          AVG(note) AS M2
6      FROM
7          retour_client
8      JOIN produit USING (cle_produit)
9      WHERE
10         date_achat BETWEEN '2021-04-01' AND '2021-06-30'
11     GROUP BY typologie_produit
12 ), T1 AS (
13     SELECT
14         typologie_produit,
15         AVG(note) AS M1
16     FROM
17         retour_client
18     JOIN produit USING (cle_produit)
19     WHERE
20         date_achat BETWEEN '2021-01-01' AND '2021-03-31'
21     GROUP BY typologie_produit
22 )
23 SELECT
24     T1.typologie_produit, ROUND((M2 - M1)/M1*100, 2) AS 'progression_%'
25 FROM
26     T1
27 JOIN T2 ON T1.typologie_produit = T2.typologie_produit
28 WHERE
29     (M2 - M1) > 0;
```

	typologie_produit	progression_%
▶	Alimentaire	0.81
	Loisirs	4.24

Les loisirs, meilleure progression

4) Requêtes SQL et Analyses

```
1  -- Net Promoter Score
2  • WITH T2 AS
3  (SELECT
4      COUNT(*) AS C2
5      FROM
6      retour_client
7      WHERE
8      note >= 9) ,
9  T1 AS
10 (SELECT
11     COUNT(*) AS C1
12     FROM
13     retour_client
14     WHERE
15     note <= 6
16 ),
17 T0 AS
18 (SELECT
19     COUNT(*) AS C0
20     FROM
21     retour_client
22 )
23 SELECT ROUND((C2-C1)/C0*100, 2) AS 'NPS' FROM T0, T1, T2 ;
```

	NPS
▶	30.97

Techniquement, tout score supérieur à 0 peut être considéré comme un « bon » score.

Cela signifie en effet que vous avez plus de promoteurs que de détracteurs. Selon les normes NPS internationales, un score supérieur à 50 est bon, et un score supérieur à 70 est exceptionnel. Cependant, ces deux scores sont rares.

Sources : <https://www.salesforce.com/fr/learning-centre/customer-service/calculate-net-promoter-score/>

4) Requêtes SQL et Analyses

```
1  -- NPS par source
2  • WITH T2 AS
3  (SELECT libelle_source, COUNT(*) AS C2
4     FROM
5     retour_client
6     WHERE
7     note >= 9
8     GROUP BY libelle_source
9     ),
10  T1 AS
11  (SELECT libelle_source, COUNT(*) AS C1
12     FROM
13     retour_client
14     WHERE
15     note <= 6
16     GROUP BY libelle_source
17     ),
18  T0 AS
19  (SELECT libelle_source, COUNT(*) AS C0
20     FROM
21     retour_client
22     GROUP BY libelle_source
23     )
24  SELECT T0.libelle_source, ROUND((C2-C1)/C0*100, 2) AS 'NPS'
25  FROM T0
26  JOIN T1 USING(libelle_source)
27  JOIN T2 USING(libelle_source);
```

	libelle_source	NPS
▶	email	29.65
	téléphone	33.81
	réseaux sociaux	29.56

Le meilleur score pour le contact téléphonique

4) Requêtes SQL et Analyses

```
1 -- Quel est le nombre de retour clients par source ?
2 • SELECT
3     libelle_source, COUNT(*) AS 'nombre_retour_clients'
4 FROM
5     retour_client
6 GROUP BY libelle_source
7 ORDER BY nombre_retour_clients DESC;
```

	libelle_source	nombre_retour_clients
►	email	1032
	réseaux sociaux	998
	téléphone	970

```
1 -- Quels sont les 5 magasins avec le plus de feedbacks ?
2 • SELECT
3     ref_magasin,
4     departement,
5     libelle_de_commune,
6     COUNT(*) AS 'nb_retours'
7 FROM
8     retour_client
9     JOIN
10    ref_magasin USING(ref_magasin)
11 WHERE
12     libelle_categorie = 'expérience en magasin'
13 GROUP BY ref_magasin
14 ORDER BY nb_retours DESC
15 LIMIT 5;
```

	ref_magasin	departement	libelle_de_commune	nb_retours
►	50	78	Versailles	14
	5	94	Villecresnes	14
	44	94	Fontenay-sous-Bois	12
	22	95	Saint-Brice-sous-Forêt	11
	35	95	Saint-Gratien	11

Plusieurs contrôles de cohérence de la base des données peuvent être réalisés :

```
1 -- Repérage des doublons table retour_client
2 • SELECT
3     cle_retour_client, COUNT(*) AS nombre_de_doublons
4 FROM
5     retour_client
6 GROUP BY cle_retour_client
7 HAVING COUNT(*) > 1;
```

- *localisation des valeurs NULL*

[illegible]

5) Cohérence des données

- *contrôle des valeurs distinctes*

```
1  -- Contrôle des valeurs différentes de la colonne libelle_source
2  • SELECT DISTINCT
3      libelle_source,
4      COUNT(*) AS nb,
5      ROUND((COUNT(*) / (SELECT
6          COUNT(*)
7          FROM
8          retour_client)) * 100,
9          2) AS pourcentage
10 FROM
11     retour_client
12 GROUP BY libelle_source;
```

	libelle_source	nb	pourcentage
▶	téléphone	970	32.33
	email	1032	34.40
	réseaux sociaux	998	33.27

Pas d'espaces invisibles ou d'erreur de saisie

- *contrôle des bornes de valeurs*

```
1  -- Contrôle des bornes de valeurs de note
2  • SELECT
3      MIN(note) AS min_note, MAX(note) AS max_note
4  FROM
5      retour_client;
```

	min_note	max_note
▶	0	10

Pas de note négative ou de note > 10, c'est rassurant