Test Global Data Analyst - Niveau Senior (3/4)

Contexte: Entretien pour Data Analyst Senior - UniBank Haiti
Durée estimée: 60-75 minutes
Nombre de questions: 35

Section A: Analyse Avancée (10 questions)

Q1. Vous devez analyser l'impact d'une nouvelle politique de crédit. Comment concevriez-vous une étude A/B test pour mesurer son efficacité?

R1. Conception d'un A/B test: 1. **Hypothèse:** La nouvelle politique réduit le taux de défaut 2. **Groupes:** - Contrôle: Ancienne politique - Test: Nouvelle politique 3. **Randomisation:** Attribution aléatoire des nouvelles demandes 4. **Taille d'échantillon:** Calcul basé sur: - Effet attendu (ex: réduction de 5% à 4%) - Puissance souhaitée (80%) - Niveau de confiance (95%) 5. **Métriques:** NPL ratio, taux d'approbation, temps de décision 6. **Durée:** Suffisante pour observer des défauts (12-18 mois) 7. **Analyse:** Test z de proportion, intervalles de confiance

---

Q2. Expliquez le concept de multicolinéarité et son impact sur une régression. Comment la détectez-vous et la traitez-vous?

R2. Multicolinéarité: Forte corrélation entre variables indépendantes.

Impact: - Coefficients instables - Intervalles de confiance élargis - Interprétation difficile - Prédictions toujours valides

Détection: - Matrice de corrélation ($r > 0.7$) - VIF (Variance Inflation Factor) > 5 ou 10 - Condition number > 30

Traitement: - Supprimer une des variables corrélées - Combiner les variables (moyenne, PCA) - Régularisation (Ridge, Lasso)

```
from statsmodels.stats.outliers_influence import variance_inflation_factor
vif = [variance_inflation_factor(X.values, i) for i in range(X.shape[1])]
```

Q3. Comment valideriez-vous un modèle de scoring de crédit? Quelles métriques utiliseriez-vous?

R3. Métriques de discrimination: - **AUC-ROC:** Capacité à distinguer bons/mauvais (> 0.7 acceptable) - **Gini:** 2×AUC - 1 (> 0.4 acceptable) - **KS (Kolmogorov-Smirnov):** Séparation maximale des distributions

Métriques de calibration: - **Hosmer-Lemeshow:** Adéquation probabilités prédites vs observées - **Courbe de calibration:** Graphique proba prédite vs réelle

Stabilité: - **PSI (Population Stability Index):** Dérive du modèle (< 0.1 stable) - **CSI (Characteristic Stability Index):** Par variable

Validation: - Cross-validation k-fold - Out-of-time validation - Out-of-sample validation

---

Q4. Qu'est-ce que la régularisation L1 (Lasso) et L2 (Ridge)? Quand utiliser chacune?

R4. | | L1 (Lasso) | L2 (Ridge) | |-|———-|———-| | **Pénalité** | $\Sigma\lvert\beta_i\rvert$ | $\Sigma\beta_i^2$ | | **Effet** | Certains $\beta \to 0$ | β réduits mais $\neq 0$ | | **Usage** | Sélection de variables | Multicolinéarité | | **Résultat** | Modèle sparse | Tous les prédicteurs |

Elastic Net: Combinaison des deux.

Q5. Comment détecteriez-vous des anomalies/fraudes dans les transactions bancaires?

R5. Approches: 1. **Règles métier:** Montant > seuil, horaires inhabituels 2. **Statistique:** Z-score, IQR sur le comportement historique du client 3. **Machine Learning:** - Isolation Forest - One-class SVM - Autoencoders 4. **Séquences:** Patterns de transactions inhabituels 5. **Réseau:** Analyse des liens entre comptes

```
from sklearn.ensemble import IsolationForest
iso_forest = IsolationForest(contamination=0.01)
predictions = iso_forest.fit_predict(X)
anomalies = X[predictions == -1]
```

Q6. Expliquez la différence entre bias et variance dans un modèle ML. Comment trouver le bon équilibre?

R6. - **Bias (biais):** Erreur due à des hypothèses simplificatrices. Modèle sous-ajusté. - **Variance:** Sensibilité aux fluctuations des données. Modèle sur-ajusté.

Trade-off: - Modèle simple → High bias, low variance - Modèle complexe → Low bias, high variance

Équilibre: - Cross-validation - Régularisation - Early stopping - Plus de données - Feature engineering

---

Q7. Comment géreriez-vous un dataset fortement déséquilibré (ex: 2% de défauts)?

R7. Techniques: 1. **Resampling:** - Oversampling minoritaire (SMOTE) - Undersampling majoritaire - Combinaison 2. **Pondération:** class_weight='balanced' 3. **Métriques adaptées:** Precision, Recall, F1, AUC (pas accuracy) 4. **Seuil de décision:** Ajuster le threshold de classification 5. **Algorithmes robustes:** Random Forest, XGBoost avec paramètres adaptés

```
from imblearn.over_sampling import SMOTE
smote = SMOTE(random_state=42)
X_resampled, y_resampled = smote.fit_resample(X, y)
```

Q8. Qu'est-ce qu'une analyse de cohorte? Donnez un exemple bancaire.

R8. **Définition:** Suivi longitudinal de groupes définis par une caractéristique commune (généralement la date d'un événement).

Exemple bancaire - Rétention: - Cohorte = Mois d'ouverture du compte - Suivi = Activité mensuelle - Métrique = % clients actifs

	Mois 0	Mois 1	Mois 2	Mois 3
Jan-24	100%	85%	78%	72%
Feb-24	100%	88%	80%	-
Mar-24	100%	82%	-	-

Autres applications: - Performance des prêts (vintage analysis) - LTV par cohorte d'acquisition - Adoption de produits digitaux

---

Q9. Comment calculeriez-vous le Customer Lifetime Value (CLV) pour une banque?

R9. Formule simplifiée:

```
CLV = Revenu Moyen Annuel × Durée Relation × Marge

Exemple:
Revenu annuel = 5,000 HTG (frais + marge intérêts)
Durée moyenne = 7 ans
Marge nette = 40%
CLV = 5,000 × 7 × 0.4 = 14,000 HTG
```

Formule avec discount:

```
CLV = Σ (Revenu_t - Coût_t) / (1 + r)^t
```

Approche probabiliste:

```
CLV = m × r / (1 + d - r)
m = marge par période
r = taux de rétention
d = taux de discount
```

Q10. Expliquez le concept de time series forecasting. Quels modèles utiliseriez-vous pour prédire les dépôts mensuels?

R10. Composantes d'une série temporelle: - **Tendance:** Direction générale - **Saisonnalité:** Patterns récurrents - **Cyclicité:** Fluctuations longue période - **Bruit:** Variations aléatoires

Modèles: 1. **ARIMA:** Autorégressif + Moyenne mobile 2. **SARIMA:** ARIMA + Saisonnalité 3. **Exponential Smoothing:** Holt-Winters 4. **Prophet (Facebook):** Tendance + Saisonnalité + Holidays 5. **LSTM:** Deep learning pour séquences

```python
from statsmodels.tsa.holtwinters import ExponentialSmoothing
model = ExponentialSmoothing(data, seasonal='add', seasonal_periods=12)
forecast = model.fit().forecast(12)
```

Section B: SQL Avancé (8 questions)

Q11. Écrivez une requête pour calculer la moyenne mobile sur 6 mois du volume de transactions.

R11.

```sql
SELECT
    mois,
    volume,
    AVG(volume) OVER (
        ORDER BY mois
        ROWS BETWEEN 5 PRECEDING AND CURRENT ROW
    ) as mm6
FROM monthly_transactions;
```

Q12. Comment implémenteriez-vous une analyse de gaps (périodes sans activité) en SQL?

R12.

```sql
WITH activity AS (
    SELECT
        client_id,
        DATE(date_tx) as date_activite,
        LAG(DATE(date_tx)) OVER (
            PARTITION BY client_id ORDER BY date_tx
        ) as date_precedente
    FROM transactions
),
gaps AS (
    SELECT
        client_id,
        date_precedente as debut_gap,
        date_activite as fin_gap,
        date_activite - date_precedente as duree_gap
    FROM activity
    WHERE date_activite - date_precedente > 30
)
SELECT * FROM gaps ORDER BY duree_gap DESC;
```

Q13. Créez une requête recursive pour calculer le solde cumulé jour par jour.

R13.

```sql
WITH RECURSIVE daily_balance AS (
    -- Cas de base
    SELECT
        compte_id,
        date_tx,
        montant,
        montant as solde_cumule
    FROM transactions
    WHERE date_tx = (SELECT MIN(date_tx) FROM transactions)

    UNION ALL

    -- Récursion
    SELECT
        t.compte_id,
        t.date_tx,
        t.montant,
        db.solde_cumule + t.montant
    FROM transactions t
    JOIN daily_balance db ON t.compte_id = db.compte_id
    WHERE t.date_tx = db.date_tx + INTERVAL '1 day'
)
SELECT * FROM daily_balance;
```

Alternative plus simple avec window function:

```sql
SELECT
    date_tx,
    montant,
    SUM(montant) OVER (ORDER BY date_tx) as solde_cumule
FROM transactions;
```

Q14. Comment optimiseriez-vous une requête qui joint plusieurs grandes tables?

R14. 1. **Index:** Créer des index sur les colonnes de jointure 2. **Filtrer tôt:** Appliquer les WHERE avant les JOIN 3. **Sélection:** Éviter SELECT *, spécifier les colonnes 4. **Partitionnement:** Tables partitionnées par date 5. **EXPLAIN ANALYZE:** Analyser le plan d'exécution 6. **Statistiques:** Mettre à jour avec ANALYZE

```sql
-- Mauvais
SELECT * FROM a JOIN b ON a.id = b.id JOIN c ON b.id = c.id WHERE ...;

-- Bon
WITH filtered_a AS (
    SELECT id, col1, col2 FROM a WHERE date > '2024-01-01'
)
SELECT fa.col1, b.col2, c.col3
FROM filtered_a fa
JOIN b ON fa.id = b.a_id
JOIN c ON b.id = c.b_id;
```

Q15. Écrivez une requête pour détecter les tendances (croissance/décroissance) sur 3 mois consécutifs.

R15.

```sql
WITH monthly AS (
    SELECT
        DATE_TRUNC('month', date) as mois,
        SUM(montant) as total
    FROM transactions
    GROUP BY DATE_TRUNC('month', date)
),
trends AS (
    SELECT
        mois,
        total,
        LAG(total, 1) OVER (ORDER BY mois) as m1,
        LAG(total, 2) OVER (ORDER BY mois) as m2
    FROM monthly
)
SELECT
    mois,
    total,
    CASE
        WHEN total > m1 AND m1 > m2 THEN 'Croissance 3 mois'
        WHEN total < m1 AND m1 < m2 THEN 'Décroissance 3 mois'
        ELSE 'Stable/Variable'
    END as tendance
FROM trends
WHERE m1 IS NOT NULL AND m2 IS NOT NULL;
```

Q16-18. [Questions supplémentaires SQL avancé...]

Section C: DAX Avancé (7 questions)

Q19. Créez une mesure DAX pour calculer le taux de rétention client mensuel.

R19.

```
Taux Retention =
VAR ClientsDebutMois = CALCULATE(
    DISTINCTCOUNT(Clients[ClientID]),
    PREVIOUSMONTH(Calendrier[Date])
)
VAR ClientsPerdus = CALCULATE(
    DISTINCTCOUNT(Clients[ClientID]),
    Clients[Statut] = "Fermé",
    DATESMTD(Calendrier[Date])
)
RETURN
DIVIDE(ClientsDebutMois - ClientsPerdus, ClientsDebutMois)
```

Q20. Implémentez un calcul de percentile en DAX.

R20.

```
P90 Montant =
PERCENTILE.INC(Transactions[Montant], 0.9)

// Alternative avec CALCULATE pour contexte
P90 par Agence =
CALCULATE(
    PERCENTILE.INC(Transactions[Montant], 0.9),
    ALLEXCEPT(Transactions, Agences[Agence])
)
```

Q21-25. [Questions supplémentaires DAX...]

Section D: Cas Pratique (10 questions)

Q26. Le CEO demande: "Pourquoi notre NPL a augmenté de 0.5% ce trimestre?" Comment structureriez-vous votre analyse?

R26. Framework d'analyse:

1. **Décomposition du NPL:**
   - Par vintage (cohorte d'octroi)
   - Par secteur économique
   - Par agence/région
   - Par type de produit
2. **Analyse des drivers:**
   - Nouveaux défauts vs aggravation
   - Migration entre buckets (30j → 60j → 90j)
   - Concentration (grands expositions)
3. **Facteurs externes:**
   - Conditions économiques

- Événements sectoriels (sécheresse pour agriculture)
- Évolution des taux

4. **Facteurs internes:**
   - Changements de politique de crédit
   - Qualité du scoring
   - Capacité de recouvrement
5. **Présentation:**
   - Executive summary
   - Waterfall chart de l'évolution NPL
   - Recommandations actionables

Q27-35. [Questions supplémentaires cas pratiques...]

Scoring

Score	Niveau
0-12	À améliorer
13-20	Intermédiaire
21-28	Avancé
29-33	Senior
34-35	Expert/Lead