

Manuel de Révision Complet - Data Analyst UniBank Haiti

Plan de Révision (Dernière Journée)

| Heure | Activité | Durée |
|-------|------------------------------|--------|
| 08:00 | Statistiques et Probabilités | 1h30 |
| 09:30 | Pause | 15 min |
| 09:45 | SQL et DAX | 1h30 |
| 11:15 | Pause | 15 min |
| 11:30 | BI Bancaire et KPIs | 1h |
| 12:30 | Déjeuner | 45 min |
| 13:15 | Python et Visualisation | 1h |
| 14:15 | Pause | 15 min |
| 14:30 | Études de Cas | 1h30 |
| 16:00 | Fiches de Synthèse | 30 min |

1. STATISTIQUES ESSENTIELLES

1.1 Statistiques Descriptives - À Retenir

Mesures de tendance centrale: - **Moyenne:** Sensible aux outliers, utiliser quand données symétriques - **Médiane:** Robuste, préférer quand outliers ou asymétrie - **Mode:** Pour variables catégorielles

Mesures de dispersion: - **Écart-type:** Interprétable dans les mêmes unités - **Variance:** Carré de l'écart-type - **IQR:** Q3 - Q1, robuste pour détecter outliers - **CV:** Permet de comparer des dispersions différentes

Règle 68-95-99.7 (Distribution normale):

68% dans ± 1
95% dans ± 2
99.7% dans ± 3

1.2 Tests d'Hypothèses - Procédure

1. Formuler H_0 et H_a
2. Choisir α (généralement 0.05)
3. Calculer la statistique de test
4. Calculer la p-value
5. Si p-value $< \alpha \rightarrow$ Rejeter H_0
6. Interpréter dans le contexte

Choix du test: | Comparer | Test Paramétrique | Non-Paramétrique | |-----|-----|-----|
|-----| | 1 moyenne vs valeur | t-test 1 sample | Wilcoxon signed-rank | | 2 moyennes indép.
| t-test indépendant | Mann-Whitney U | | 2 moyennes appariées | t-test apparié | Wilcoxon
signed-rank | | 3+ moyennes | ANOVA | Kruskal-Wallis | | 2 proportions | z-test | Chi-carré | |
Indépendance | - | Chi-carré |

1.3 Corrélation - Points Clés

Pearson (r): Relation linéaire, données continues, normales

Spearman (): Relation monotone, ordinales ou non-linéaires

Interprétation |r|:

< 0.3: Faible

0.3-0.7: Modérée

> 0.7: Forte

ATTENTION: Corrélation Causalité

1.4 Probabilités - Formules

Bayes: $P(A|B) = [P(B|A) \times P(A)] / P(B)$

Distributions:

- Binomiale: $P(X=k) = C(n,k) \times p^k \times (1-p)^{(n-k)}$

- Poisson: $P(X=k) = (\lambda^k \times e^{-\lambda}) / k!$

- Normale: $Z = (X - \mu) / \sigma$

2. SQL POUR DATA ANALYST

2.1 Window Functions - Synthèse

-- Classement

ROW_NUMBER() -- Numéro unique

RANK() -- Saute si égalité (1,2,2,4)

DENSE_RANK() -- Ne saute pas (1,2,2,3)

NTILE(n) -- Divise en n groupes

-- Navigation

LAG(col, n) -- n lignes avant

LEAD(col, n) -- n lignes après

FIRST_VALUE(col) -- Première valeur

LAST_VALUE(col) -- Dernière valeur

-- Agrégation

SUM() OVER (ORDER BY date) -- Cumul

AVG() OVER (ROWS BETWEEN 6 PRECEDING AND CURRENT ROW) -- MA 7

2.2 CTE et Sous-requêtes

-- CTE simple

WITH stats AS (

SELECT agence, SUM(montant) as total

FROM transactions

GROUP BY agence

)

SELECT * FROM stats WHERE total > 100000;

-- CTE multiple

WITH

cte1 AS (SELECT ...),

```
cte2 AS (SELECT ... FROM cte1 ...)
SELECT ... FROM cte2;
```

2.3 Patterns Utiles

```
-- Top N par groupe
WITH ranked AS (
    SELECT *, ROW_NUMBER() OVER (
        PARTITION BY agence ORDER BY solde DESC
    ) as rn
    FROM clients
)
SELECT * FROM ranked WHERE rn <= 5;

-- YoY Comparison
SELECT
    mois,
    total,
    LAG(total, 12) OVER (ORDER BY mois) as total_n1,
    (total - LAG(total, 12) OVER (ORDER BY mois)) /
    NULLIF(LAG(total, 12) OVER (ORDER BY mois), 0) * 100 as var_pct
FROM monthly_sales;

-- Cumul running
SELECT
    date,
    montant,
    SUM(montant) OVER (ORDER BY date) as cumul
FROM transactions;
```

2.4 Optimisation

Bonnes pratiques:

- SELECT colonnes spécifiques
- Index sur colonnes WHERE, JOIN, ORDER BY
- EXISTS plutôt que IN pour sous-requêtes
- LIMIT pour limiter les résultats

À éviter:

- SELECT *
- Fonctions sur colonnes indexées dans WHERE
- OR sur colonnes différentes
- Sous-requêtes corrélées si possible

3. DAX - RÉSUMÉ

3.1 Contextes

Contexte de LIGNE (Row): Colonne calculée

- Accède aux valeurs de la ligne courante
- Calculé au refresh, stocké

Contexte de FILTRE (Filter): Mesure
- Influencé par slicers, filtres, visuels
- Calculé dynamiquement

3.2 CALCULATE - Le Cœur de DAX

```
CALCULATE(expression, filtre1, filtre2, ...)  
  
// Modificateurs importants:  
ALL(Table)           -- Supprime tous les filtres  
ALLEXCEPT(T, Col)  -- Garde certains filtres  
FILTER(T, cond)       -- Table filtrée  
KEEPFILTERS(cond)    -- Ajoute sans remplacer
```

3.3 Time Intelligence

```
// To-Date  
TOTALYTD(mesure, Date)  
TOTALQTD(mesure, Date)  
TOTALMTD(mesure, Date)  
  
// Périodes précédentes  
SAMEPERIODLASTYEAR(Date)  
PREVIOUSMONTH(Date)  
PREVIOUSYEAR(Date)  
  
// Glissant  
DATESINPERIOD(Date, MAX(Date), -12, MONTH)  
DATESBETWEEN(Date, debut, fin)  
  
// Pattern Variation YoY  
Var YoY =  
VAR Actuel = SUM(Ventes[Montant])  
VAR AnPrec = CALCULATE(SUM(Ventes[Montant]),  
                        SAMEPERIODLASTYEAR(Cal[Date]))  
RETURN DIVIDE(Actuel - AnPrec, AnPrec)
```

3.4 Patterns Courants

```
// % du Total  
% Total = DIVIDE(  
    SUM(T[Montant]),  
    CALCULATE(SUM(T[Montant]), ALL(T))  
)  
  
// Cumul  
Cumul = CALCULATE(  
    SUM(T[Montant]),  
    FILTER(ALLSELECTED(Cal[Date]), Cal[Date] <= MAX(Cal[Date]))  
)  
  
// Moyenne Mobile  
MM3 = AVERAGEX(  
    DATESINPERIOD(Cal[Date], MAX(Cal[Date]), -3, MONTH),
```

```

    CALCULATE(SUM(T[Montant]))
)

// Ranking
Rang = RANKX(ALL(Clients), [Total Ventes],, DESC, Dense)

```

4. KPIs BANCAIRES

4.1 Rentabilité

| KPI | Formule | Benchmark |
|------------|---------------------------------------|-----------|
| ROE | Résultat Net / Capitaux Propres | 12-18% |
| ROA | Résultat Net / Total Actifs | 1-2% |
| NIM | (Rev. Int. - Ch. Int.) / Actifs Prod. | 3-5% |
| CIR | Charges Exploit. / PNB | 45-55% |

4.2 Qualité des Actifs

| KPI | Formule | Benchmark |
|---------------------|---------------------------|-----------|
| NPL Ratio | Prêts > 90j / Total Prêts | < 5% |
| Coverage | Provisions / NPL | > 100% |
| Cost of Risk | Dotations Prov. / Encours | 1-3% |

4.3 Solvabilité et Liquidité

| KPI | Formule | Exigence |
|------------|---------------------|-------------|
| CAR | Fonds Propres / RWA | ≥ 12% (BRH) |
| LDR | Prêts / Dépôts | 80-90% |
| LCR | HQLA / Sorties 30j | ≥ 100% |

4.4 Commercial

| KPI | Formule | Usage |
|-------------------|--------------------------------------|----------------------|
| Cross-sell | Nb Produits / Nb Clients | Engagement |
| Churn | Clients Perdus / Clients Début | Rétention |
| CAC | Coûts Acquisition / Nouveaux Clients | Efficacité marketing |
| LTV | Revenu × Durée × Marge | Valeur client |
| NPS | % Promoteurs - % Détracteurs | Satisfaction |

5. PYTHON - RAPPELS

5.1 Pandas Essentiels

```

# Chargement et exploration
df = pd.read_csv('file.csv')

```

```

df.head(), df.info(), df.describe()
df.shape, df.columns, df.dtypes

# Valeurs manquantes
df.isnull().sum()
df.fillna(df['col'].median())
df.dropna(subset=['col'])

# Filtrage
df[df['col'] > 100]
df[(cond1) & (cond2)]
df.query('col > 100 and type == "A"')

# Agrégation
df.groupby('cat')['val'].agg(['sum', 'mean', 'count'])
df.pivot_table(values='val', index='row', columns='col', aggfunc='sum')

# Transformation
df['new'] = df['a'] / df['b']
df['cat'] = pd.cut(df['val'], bins=[0, 100, 500, 1000])
df['date'] = pd.to_datetime(df['date'])
df['year'] = df['date'].dt.year

```

5.2 Visualisation (Matplotlib/Seaborn)

```

import matplotlib.pyplot as plt
import seaborn as sns

# Histogramme
plt.hist(df['col'], bins=30)

# Box plot
df.boxplot(column='val', by='cat')

# Scatter
plt.scatter(df['x'], df['y'])

# Heatmap corrélation
sns.heatmap(df.corr(), annot=True, cmap='coolwarm')

```

6. EDA - MÉTHODOLOGIE

6.1 Framework

1. COMPRENDRE le contexte business
2. CHARGER et examiner la structure
3. PROFILER chaque variable (univarié)
4. EXPLORER les relations (bivarié)
5. IDENTIFIER problèmes de qualité
6. NETTOYER et transformer
7. DOCUMENTER les insights

6.2 Checklist Qualité Données

- Valeurs manquantes (isnull)
- Doublons (duplicated)
- Types de données corrects
- Valeurs aberrantes (outliers)
- Cohérence (cross-validation)
- Distributions attendues
- Cardinalité des catégories

6.3 Traitement des Outliers

```
# Méthode IQR
Q1, Q3 = df['col'].quantile([0.25, 0.75])
IQR = Q3 - Q1
lower = Q1 - 1.5 * IQR
upper = Q3 + 1.5 * IQR
outliers = df[(df['col'] < lower) | (df['col'] > upper)]

# Méthode Z-score
from scipy import stats
z = np.abs(stats.zscore(df['col']))
outliers = df[z > 3]
```

7. SEGMENTATION CLIENT

7.1 RFM

R (Recency): Jours depuis dernière activité
F (Frequency): Nombre de transactions
M (Monetary): Montant total

Score 1-5 par quintile, inversé pour R

7.2 Segments Types

| Segment | Profil | Action |
|-----------|----------------|------------------------|
| Champions | RFM élevé | Fidéliser, récompenser |
| Fidèles | F+M élevé | Maintenir, cross-sell |
| Nouveaux | R élevé, F bas | Activer, onboarding |
| À risque | R bas, F élevé | Réactiver |
| Perdus | Tout bas | Win-back sélectif |

8. FORMULES IMPORTANTES

Statistiques

Moyenne: $\bar{x} = \sum x / n$

Variance: $s^2 = \sum (x - \bar{x})^2 / (n-1)$

IC 95%: $\bar{x} \pm 1.96 \times (s/\sqrt{n})$

Finance

ROE = Résultat / Capitaux Propres

Expected Loss = PD \times LGD \times EAD

CAGR = $(V_f/V_i)^{(1/n)} - 1$

Variation

Var % = $(\text{Nouveau} - \text{Ancien}) / \text{Ancien} \times 100$

YoY = $(\text{Année N} - \text{Année N-1}) / \text{Année N-1} \times 100$

9. CONSEILS POUR L'ENTRETIEN

Questions Techniques

1. Toujours donner un exemple concret (bancaire si possible)
2. Expliquer le "pourquoi" pas juste le "quoi"
3. Mentionner les limites et alternatives

Études de Cas

1. CLARIFIER le problème et les données
2. STRUCTURER l'approche avant de commencer
3. EXPLIQUER les choix méthodologiques
4. INTERPRÉTER dans le contexte business
5. PROPOSER des next steps

Communication

- Vulgariser pour non-techniques
 - Utiliser des analogies
 - Admettre ce qu'on ne sait pas
-

10. TERMES À CONNAÎTRE

| Terme | Définition Rapide |
|----------------------------|---|
| ACID | Atomicity, Consistency, Isolation, Durability |
| ETL | Extract, Transform, Load |
| OLAP | Online Analytical Processing (analyse) |
| OLTP | Online Transaction Processing (opérationnel) |
| Data Warehouse | Entrepôt de données historiques |
| Data Lake | Stockage données brutes |
| Feature Engineering | Création de variables |
| Overfitting | Modèle trop ajusté aux données d'entraînement |
| Cross-validation | Validation croisée |
| p-value | Probabilité d'observer le résultat si H_0 vraie |

CHECKLIST FINALE

Statistiques descriptives et tests
SQL: CTEs, Window Functions, optimisation
DAX: CALCULATE, Time Intelligence, contextes
KPIs bancaires (rentabilité, risque, liquidité)
Python/Pandas: manipulation, visualisation
EDA: méthodologie, qualité données
Segmentation RFM
Interprétation dans contexte business

Bonne chance pour votre entretien!