**Méthodologie pour Travailler sur le Projet Black Phoenix Data Analysis**

\*\*Objectif du Projet\*\*

Le projet vise à créer un portfolio complet qui démontre tes compétences en data science et organisation de projets analytiques, tout en s'appuyant sur une architecture bien structurée. L'objectif est de présenter des analyses, modélisations, visualisations et résultats de manière professionnelle.

### \*\*Structure et Rôle des Dossiers\*\*

Chaque dossier joue un rôle spécifique dans l'organisation du projet. Voici une explication détaillée :

#### \*\*1. « Data » : Fichiers de Données\*\*

- \*\*Contenu\*\* :

- `raw/` : Données brutes non modifiées provenant des sources originales.

- `processed/` : Données nettoyées, enrichies ou transformées pour analyse.

- `results/` : Résultats intermédiaires comme des agrégats, regroupements ou décompositions.

- \*\*Utilisation\*\* :

- Importe les données à partir de `raw/` dans tes scripts.

- Sauvegarde toutes tes versions nettoyées ou enrichies dans `processed/`.

- Stocke les sorties d'analyses quantitatives dans `results/` pour réutilisation dans les documents ou applications.

#### \*\*2. « Documents » : Livrables et Rapports\*\*

- \*\*Contenu\*\* :

- `index.qmd` : Page d'accueil du portfolio.

- `Doc1\_Analyse\_Exploratoire.qmd` : Analyse exploratoire des données.

- `Templates/` : Modèles pour les rapports (template Quarto, CSS).

- \*\*Utilisation\*\* :

- Rédige chaque étape de l'analyse dans un fichier `.qmd`.

- Utilise le template CSS pour harmoniser le style des rapports.

- Organise les liens entre les rapports depuis `index.qmd`.

#### \*\*3. « Scripts » : Code et Automatisation\*\*

- \*\*Contenu\*\* :

- `01\_import\_data.R` : Script d'importation des données.

- `02\_clean\_data.R` : Script de nettoyage et préparation des données.

- `03\_exploratory\_analysis.R` : Script pour les analyses exploratoires.

- `04\_segmentation.R` : Code de segmentation (K-means, CAH).

- `05\_modelisation.R` : Scripts pour la modélisation prédictive.

- `functions/` : Fonctions personnalisées réutilisables (ex. ggplot theme).

- \*\*Utilisation\*\* :

- Chaque script correspond à une étape clé du projet.

- Évite de modifier les données directement; utilise un script pour chaque transformation.

#### \*\*4. « Graphiques » : Visualisations\*\*

- \*\*Contenu\*\* :

- `eda/` : Graphiques de l'analyse exploratoire.

- `segmentation/` : Visualisations des clusters (ACP, K-means).

- `modelisation/` : Graphiques liés aux modèles prédictifs.

- \*\*Utilisation\*\* :

- Sauvegarde les visualisations dans ces dossiers pour les inclure dans tes rapports ou applications.

- Garde un nom explicite pour chaque fichier.

#### \*\*5. « Images » : Logos et Références Visuelles\*\*

- \*\*Contenu\*\* :

- `Black\_Phoenix\_Data\_Analysis\_Logo.jpg` : Logo principal.

- Autres icônes ou visuels spécifiques aux rapports.

- \*\*Utilisation\*\* :

- Inclue ces visuels dans les rapports et applications.

- Utilise un fichier CSS pour ajuster leur taille et position.

### \*\*6. « Outputs » : Synthèse et Livrables\*\*

- \*\*Contenu\*\* :

- `rapport\_final.pdf` : Rapport complet consolidé.

- `rapport\_final.html` : Version web.

- Présentations ou exports prédéfinis pour les livrables.

- \*\*Utilisation\*\* :

- Exporte les résultats présentables ici.

- Maintiens un lien clair entre les analyses et ces fichiers finaux.

### \*\*Méthodologie de Travail\*\*

1. \*\*Planification\*\*

- Analyse les besoins : Identifie les objectifs clés, les livrables attendus et les ressources disponibles.

- Définis les jeux de données pertinents : Liste les données nécessaires et leur source.

- Priorise les tâches : Classe les étapes en fonction de leur impact et complexité.

- Établis un objectif clair pour chaque phase (ex. analyse exploratoire, modélisation).

- Priorise les tâches en fonction de leur impact et de leur complexité.

2. \*\*Organisation\*\*

- Reste cohérent dans les noms des fichiers et scripts.

- Commence par l’importation et le nettoyage des données avec des scripts clairs (ég. `01\_import\_data.R`, `02\_clean\_data.R`).

- Documente toutes les étapes clés (commentaires dans le code, README).

3. \*\*Versionnage\*\*

- Effectue des commits réguliers dans GitHub pour suivre les modifications.

- Nomme les commits de manière claire (ex. « Ajout de l'analyse exploratoire »).

4. \*\*Rendu\*\*

- Utilise Quarto pour rendre des documents harmonisés.

- Relis et teste chaque rapport avant de le considérer comme terminé.

5. \*\*Validation\*\*

- Revérifie les résultats avec des scripts et tests secondaires.

- Compare les analyses préliminaires avec les résultats finaux.

### \*\*Plan Chronologique pour le Début du Projet\*\*

#### \*\*Semaine 1 : Mise en Place de l'Infrastructure\*\*

- \*\*Jour 1–2\*\* :

- Crée l'architecture de fichiers et dossiers (« Data », « Scripts », etc.).

- Initialise un dépôt Git pour le projet.

- Ajoute un README.md expliquant l'objectif global et la structure.

- \*\*Jour 3–5\*\* :

- Rassemble les données et place-les dans `Data/raw/`.

- Crée un premier template Quarto dans `Templates/` (CSS + YAML).

- Teste une compilation d'un document à partir de `index.qmd`.

#### \*\*Semaine 2 : Analyse Exploratoire des Données\*\*

- \*\*Jour 1–3\*\* :

- Développe le script `01\_import\_data.R` pour charger les données.

- Crée `02\_clean\_data.R` pour nettoyer et transformer les données.

- Documente les transformations effectuées.

- \*\*Jour 4–5\*\* :

- Lance une analyse exploratoire avec `03\_exploratory\_analysis.R`.

- Sauvegarde les graphiques et statistiques clés dans `Graphiques/eda`.

- Rédige un rapport initial dans `Doc1\_Analyse\_Exploratoire.qmd`.

#### \*\*Semaine 3 : Premiers Résultats et Feedback\*\*

- \*\*Jour 1–2\*\* :

- Partage les premiers résultats avec un mentor ou collègue pour feedback.

- Identifie les ajustements à apporter sur les données ou analyses.

- \*\*Jour 3–5\*\* :

- Prépare la segmentation des données (ébauche du script `04\_segmentation.R`).

- Défini les métriques clés pour valider les clusters.

1. \*\*Planification\*\*

- Établis un objectif clair pour chaque phase (ex. analyse exploratoire, modélisation).

- Priorise les tâches en fonction de leur impact et de leur complexité.

2. \*\*Organisation\*\*

- Reste cohérent dans les noms des fichiers et scripts.

- Documente toutes les étapes clés (commentaires dans le code, README).

3. \*\*Versionnage\*\*

- Effectue des commits réguliers dans GitHub pour suivre les modifications.

- Nomme les commits de manière claire (ex. « Ajout de l'analyse exploratoire »).

4. \*\*Rendu\*\*

- Utilise Quarto pour rendre des documents harmonisés.

- Relis et teste chaque rapport avant de le considérer comme terminé.

5. \*\*Validation\*\*

- Revérifie les résultats avec des scripts et tests secondaires.

- Compare les analyses préliminaires avec les résultats finaux.

### \*\*Plan Chronologique pour le Début du Projet\*\*

#### \*\*Semaine 1 : Mise en Place de l'Infrastructure\*\*

- \*\*Jour 1–2\*\* :

- Crée l'architecture de fichiers et dossiers (« Data », « Scripts », etc.).

- Initialise un dépôt Git pour le projet.

- Ajoute un README.md expliquant l'objectif global et la structure.

- \*\*Jour 3–5\*\* :

- Rassemble les données et place-les dans `Data/raw/`.

- Crée un premier template Quarto dans `Templates/` (CSS + YAML).

- Teste une compilation d'un document à partir de `index.qmd`.

#### \*\*Semaine 2 : Analyse Exploratoire des Données\*\*

- \*\*Jour 1–3\*\* :

- Développe le script `01\_import\_data.R` pour charger les données.

- Crée `02\_clean\_data.R` pour nettoyer et transformer les données.

- Documente les transformations effectuées.

- \*\*Jour 4–5\*\* :

- Lance une analyse exploratoire avec `03\_exploratory\_analysis.R`.

- Sauvegarde les graphiques et statistiques clés dans `Graphiques/eda`.

- Rédige un rapport initial dans `Doc1\_Analyse\_Exploratoire.qmd`.

#### \*\*Semaine 3 : Premiers Résultats et Feedback\*\*

- \*\*Jour 1–2\*\* :

- Partage les premiers résultats avec un mentor ou collègue pour feedback.

- Identifie les ajustements à apporter sur les données ou analyses.

- \*\*Jour 3–5\*\* :

- Prépare la segmentation des données (ébauche du script `04\_segmentation.R`).

- Défini les métriques clés pour valider les clusters.

### \*\*Bonnes Pratiques\*\*

- \*\*Structure logique\*\* : Respecte l'architecture du projet.

- \*\*Clarté\*\* : Documente chaque étape dans les scripts et rapports.

- \*\*Flexibilité\*\* : Sois prêt à ajuster les scripts ou données en fonction des résultats.

- \*\*Sauvegarde\*\* : Commit régulier sur GitHub pour éviter les pertes.

En suivant cette méthodologie et planification chronologique, tu pourras travailler efficacement tout en préservant la qualité et la cohérence du projet.