

AssurAnalytics

Analyse des Sinistres & Profil des Assurés

Dashboard Interactif — Python · Dash · Plotly

Auteur Sona KOULIBALY	Formation Mastère 2 Big Data & Data Stratégie
Technologie Python 3.12 — Dash — Plotly — Pandas	Année Février 2026

1.

Présentation du Projet

⌚ 1.1 Contexte & Problématique

Ce projet s'inscrit dans le cadre du Mastère 2 Big Data & Data Stratégie. Il répond à une problématique concrète rencontrée dans le secteur de l'assurance :

Comment permettre à une compagnie d'assurance de suivre ses sinistres, identifier les profils à risque, visualiser les tendances et aider à la décision pour la tarification ou la prévention ?

⌚ 1.2 Objectifs

- Fournir un tableau de bord interactif pour le suivi en temps réel des sinistres
- Identifier les profils assurés à risque élevé
- Analyser la rentabilité par type de produit et région
- Automatiser la génération d'insights et recommandations tarifaires • Permettre l'export des données filtrées en Excel, HTML et PDF

❖ 1.3 Stack Technologique

Technologie	Version	Rôle
Python	3.12	Langage principal — traitement des données
Dash	2.17.1	Framework web interactif — composants réactifs
Dash Bootstrap Components	1.6.0	UI Bootstrap — grille, cartes, boutons
Plotly	5.22.0	Moteur de graphiques interactifs
Pandas	2.2.2	Manipulation, filtrage et agrégation des données
NumPy	1.26.4	Calculs numériques et transformations
OpenPyXL	3.1.4	Génération de fichiers Excel multi-feuilles
ReportLab	4.2.2	Génération de rapports PDF
Gunicorn	22.0.0	Serveur WSGI pour déploiement production

Base de Données

📊 2.1 Description du Jeu de Données

Fichier : data/assurance_data_1000.csv — Séparateur : point-virgule (;)

2.

- 1 000 observations (assurés)
- 11 variables originales
- 4 variables calculées automatiquement à l'initialisation
- 4 régions du Sénégal : Dakar, Thiès, Kaolack, Saint-Louis
- 4 types d'assurance : Auto, Santé, Habitation, Vie

2.2 Dictionnaire des Variables

Variable	Type	Valeurs	Description
id_assure	int	1 – 1000	Identifiant unique de l'assuré
age	int	18 – 79 ans	Âge de l'assuré
sexe	str	masculin / féminin	Sexe de l'assuré
type_assurance	str	Auto / Santé / Habitation / Vie	Catégorie de produit
duree_contrat	int	1 – 10 ans	Durée du contrat (années)
montant_prime	float	100 – 600 €	Prime annuelle versée
nb_sinistres	int	0 – 4	Nombre de sinistres déclarés
montant_sinistres	float	10 – 9 988 €	Montant total des sinistres payés
date_derniere_sinistre	datetime	2020 – 2025	Date du dernier sinistre
region	str	4 régions	Région de résidence au Sénégal
bonus_malus	float	0.50 – 1.50	Coefficient bonus/malus (auto)

2.3 Variables Calculées à l'Initialisation

Variable	Calcul	Description
tranche_age	pd.cut() — 6 intervalles	Segmentation : 18-25, 26-35, 36-45, 46-55, 56-65, 66-79
ratio_SP	montant_sinistres / montant_prime	Indicateur de rentabilité — ratio > 1 = assuré déficitaire
bm_cat	pd.cut() — 4 catégories	Bonus fort / Bonus / Neutre / Malus

3.

annee_sinistre	dt.year	Année du dernier sinistre pour analyse temporelle
----------------	---------	---

Architecture Technique

3.1 Structure des Fichiers

```
assuranalytics/
    ├── app.py          # Cœur : initialisation Dash + chargement données
    ├── layout.py       # Interface : composants HTML/Dash
    ├── callbacks.py    # Logique : callbacks + graphiques + exports
    ├── requirements.txt # Dépendances Python
    ├── data/
    │   └── assurance_data_1000.csv
    ├── assets/
    │   ├── style.css    # Design personnalisé
    │   └── logo*.png
    └── docs/
        └── documentation.docx
```

3.2 Rôle de Chaque Fichier

app.py — Cœur de l'Application

- Initialise l'application Dash avec Bootstrap et FontAwesome
- Charge le fichier CSV et enrichit le DataFrame (tranche_age, ratio_SP, bm_cat)
- Déclare le layout et enregistre les callbacks
- Expose le server Gunicorn pour déploiement

layout.py — Interface Utilisateur

- Définit toute la structure HTML via des composants Dash
- Header avec logo, titre et boutons d'export
- Section insights dynamiques (storytelling)
- Panneau filtres gauche (6 filtres interactifs)
- 8 KPIs (4 principaux + 4 secondaires)
- 5 sections de graphiques (13 visualisations)
- Tableau de données interactif

callbacks.py — Logique & Interactivité

- filter_data() : fonction centrale de filtrage multi-critères
- generate_insights() : génération de 9 insights contextuels automatiques
- update_all : callback principal — 6 inputs → 30 outputs simultanés
- reset_filters : réinitialisation des 6 filtres en un clic
- download_excel : export Excel avec 4 feuilles de données
- download_html : rapport HTML avec graphiques Plotly embarqués
- download_pdf : rapport PDF structuré avec ReportLab

4.

3.3 Flux de Données

CSV → pandas DataFrame → Enrichissement (tranche_age, ratio_SP, bm_cat)

filter_data() [6 critères]

/ | \

Graphiques KPIs + Insights Exports
Plotly (13) Dynamiques (8) Excel/HTML/PDF

4. Fonctionnalités du Dashboard

🔍 4.1 Filtres Interactifs

Filtre	Type	Options
Type d'assurance	Dropdown multi-sélection	Auto / Santé / Habitation / Vie
Sexe	Dropdown multi-sélection	Masculin / Féminin
Région	Dropdown multi-sélection	Dakar / Kaolack / Saint-Louis / Thiès
Nb sinistres	Dropdown multi-sélection	0 / 1 / 2 / 3 / 4+
Tranche d'âge	RangeSlider	18 – 79 ans
Bonus/Malus	RangeSlider	0.5 – 1.5

📊 4.2 KPIs — Indicateurs Clés

8 indicateurs mis à jour dynamiquement à chaque changement de filtre :

KPIs Principaux (4 cartes colorées)

- Total Assurés — Nombre d'assurés dans la sélection
- Total Sinistres — Somme des sinistres déclarés

Graphique	Type	Insight	Ce qu'il révèle
Répartition type assurance	Donut	Comparaison	Équilibre du portefeuille 4 produits
Distribution âges par type	Histogramme	Tendance	Structure démographique 18-79 ans

Prime moy. âge & sexe	Barres groupées	Comparaison	Différences tarifaires H/F par tranche
-----------------------	-----------------	-------------	--

- Coût Moyen Sinistre — Moyenne sur les assurés sinistrés uniquement
- Prime Moyenne — Prime annuelle moyenne des assurés sélectionnés

KPIs Secondaires (4 mini-cartes)

- Taux de sinistralité — % d'assurés ayant au moins 1 sinistre
- Ratio S/P médian — Rapport sinistres/primes médian
- B/M moyen — Coefficient bonus/malus moyen
- % Assurés déficitaires — Part des assurés dont ratio_SP > 1

💡 4.3 Insights Automatiques — Storytelling

La section "INSIGHTS CLÉS" génère automatiquement jusqu'à 9 alertes contextuelles :

- Sélection active : nb assurés et % du portefeuille total
- Taux de sinistralité : comparaison vs moyenne globale avec flèche directionnelle
- Coût moyen : variation vs référence globale (+/- %)
- Ratio S/P : alerte rentabilité si > 85% d'assurés déficitaires
- Région la plus coûteuse : montant et % du total
- Tranche d'âge à risque : profil prioritaire pour la tarification
- Analyse Bonus/Malus : % en malus et B/M moyen
- Recommandation tarifaire : déclenchée si coût > 3x la prime

📈 4.4 Visualisations — 13 Graphiques

Graphique	Type	Insight	Ce qu'il révèle
Répartition régionale	Donut	Comparaison	Poids de chaque région
Sinistres par région	Barres horiz.	Comparaison	Zones géographiques à risque
Fréquence sinistres	Barres	Anomalie	% assurés sans sinistre (61,7%)
Évolution temporelle	Bar + Ligne	Tendance	Saisonnalité sur 5 ans (2020-2025)
Sinistres par âge & type	Barres groupées	Relation	Tranches d'âge les plus sinistrées
Prime vs Sinistre	Nuage de points	Relation	Assurés déficitaires (au-dessus diag.)
Coût vs Prime par type	Barres groupées	Comparaison	Rentabilité par produit
Heatmap risque Âge × Type	Carte de chaleur	Anomalie	Profils à haut risque identifiés
Distribution Bonus/Malus	Donut	Tendance	Équilibre B/M du portefeuille
B/M × Sinistres × Montant	Nuage de points	Corrélation	Détection profils extrêmes

5.

Bonnes Pratiques de Visualisation

⌚ 5.1 Les 5 Insights Visuels Appliqués

Type d'Insight	Application dans AssurAnalytics
Tendance	Série temporelle mensuelle 2020–2025 : évolution du nb de sinistres et montants
Comparaison	Barres groupées H/F par tranche d'âge, coût vs prime par type d'assurance
Anomalie	Heatmap risque : détection des combinaisons âge/type les plus sinistrées
Relation	Scatter prime vs sinistre : identification des assurés déficitaires
Corrélation	Nuage B/M × sinistres × montant : corrélation coefficient et sinistralité

❖ 5.2 Qualité UX/UI

- Code couleur cohérent : bleu (info) / orange (alerte) / rouge (danger) / vert (succès)
- Filtres sticky : le panneau de filtres reste visible en scrollant
- Tooltips : info-bulles sur chaque KPI card
- Compteur dynamique : "X assurés filtrés / 1000" mis à jour en temps réel
- Mise en surbrillance conditionnelle dans le tableau (rouge si nb_sinistres > 2)
- Annotations sur les graphiques : lignes de moyenne, seuil d'équilibre, seuil malus
- Description sous chaque graphique : résumé de l'insight principal
- 5 sections titrées : navigation claire dans le dashboard

6.

↳ 5.3 Performance & Robustesse

- Callback unique centralisé : 1 callback principal avec 6 inputs → 30 outputs
- Fonction filter_data() réutilisable : un seul point de filtrage partagé par tous les callbacks
- Gestion des erreurs : empty_fig() affichée si aucune donnée ne correspond
- Vérification de nullité : toutes les agrégations vérifient len() avant calcul
- suppress_callback_exceptions=True : évite les erreurs au chargement initial

Exports & Rapports

▀ 6.1 Export Excel

- Téléchargement direct sur le PC de l'utilisateur
- 4 feuilles structurées :
 - Données — toutes les colonnes filtrées (id, âge, sexe, type, région, montant, B/M...)
 - KPIs — 8 indicateurs clés de la sélection
 - Par Région — agrégats (nb assurés, sinistres, montant, prime moy.)
 - Par Type — agrégats avec ratio S/P médian par produit
- Nom de fichier horodaté : assuranalytics_YYYYMMDD_HHMM.xlsx

🌐 6.2 Export HTML

- Rapport HTML autonome avec graphiques Plotly interactifs embarqués
- 3 graphiques inclus : Répartition type, Montants par région, Sinistres par tranche
- Bloc KPIs et insights textuels
- CSS inline — fonctionne sans connexion internet
- Nom de fichier horodaté : rapport_assuranalytics_YYYYMMDD_HHMMSS.html

7.

□ 6.3 Export PDF

- Rapport PDF structuré généré avec la librairie ReportLab
- Page de garde avec titre et date de génération
- Tableau des KPIs (8 indicateurs)
- Tableau agrégat par région
- Section insights & recommandations textuelles
- Pied de page professionnel avec copyright
- Nom de fichier horodaté : rapport_assuranalytics_YYYYMMDD_HHMMSS.pdf

Installation & Déploiement

💻 7.1 Installation Locale

1. Cloner le dépôt : `git clone https://github.com/<username>/assuranalytics.git`
2. Créer un environnement virtuel : `python -m venv venv`
3. Activer l'environnement : `venv\Scripts\activate` (Windows) ou `source venv/bin/activate` (Mac/Linux)
4. Installer les dépendances : `pip install -r requirements.txt`
5. Lancer l'application : `python app.py`
6. Ouvrir le navigateur : `http://127.0.0.1:9753`

🌐 7.2 Déploiement Production (Gunicorn)

```
gunicorn app:server --workers 4 --bind 0.0.0.0:8080
```

8.

Cloud 7.3 Plateformes de Déploiement Cloud Recommandées

- Render.com — gratuit, déploiement GitHub direct
- Railway.app — simple, PostgreSQL intégré
- Heroku — référence pour apps Python
- Azure App Service / AWS Elastic Beanstalk — pour environnements entreprise

9.

Pistes d'Amélioration

8.1 Enrichissement des Données

- Score de risque ML — modèle de régression logistique ou Random Forest pour prédire la sinistralité
- Segmentation K-Means — clustering automatique pour identifier des groupes d'assurés homogènes
- Détection d'anomalies — Isolation Forest pour repérer les profils atypiques

8.2 Nouvelles Visualisations

- Carte choroplète du Sénégal — visualisation géographique des sinistres par région
- Arbre de décision — visualisation interactive des règles de tarification
- Courbe ROC — si modèle ML de prédiction ajouté

8.3 Fonctionnalités Applicatives

- Multi-page Dash — séparer les sections en onglets dédiés (Profil / Sinistres / Rentabilité)
- Authentification — login sécurisé pour accès au dashboard
- Alertes email — notification automatique quand un profil dépasse un seuil de risque • Comparaison de périodes — A vs B (ex: 2023 vs 2024)

8.4 Infrastructure

- Base de données — migration vers PostgreSQL ou SQLite pour données en temps réel
- API REST — exposer les KPIs via FastAPI pour intégration externe
- Tests unitaires — pytest sur les fonctions filter_data() et generate_insights()

10.

- CI/CD — GitHub Actions pour déploiement automatique

11.

Captures d'Écran Recommandées pour docs/screenshots/

Pour valoriser ce projet sur GitHub et votre CV, voici les captures conseillées :

Fichier	Contenu à capturer
01_dashboard_overview.png	Vue d'ensemble complète du dashboard (zoom out navigateur à 75%)
02_header_kpis.png	Header + 8 KPIs + section insights clés
03_filtres_actifs.png	Panneau filtres avec une sélection active (ex : Auto + Dakar)
04_insights_storytelling.png	Section insights avec plusieurs alertes visibles
05_section1_profil.png	Section 1 : donut type + histogramme âges + barres groupées H/F
06_section2_sinistres.png	Section 2 : bar région + fréquence sinistres + série temporelle
07_section3_rentabilite.png	Section 3 : scatter prime/sinistre + barres coût vs prime
08_section4_risque.png	Section 4 : heatmap risque + donut B/M + nuage de points B/M

12.

09_tableau_donnees.png	Tableau interactif avec surbrillance rouge des profils à risque
10_export_excel.png	Aperçu du fichier Excel téléchargé (4 feuilles)

Conseil GitHub : Ajoutez les captures dans README.md avec la syntaxe :
![Dashboard Overview](docs/screenshots/01_dashboard_overview.png)
Cela rend le projet immédiatement attractif pour les recruteurs qui visitent votre profil.

Auteur & Portfolio GitHub

10.1 Auteur

Nom	Sona KOULIBALY
Formation	Mastère 2 Big Data & Data Stratégie
Projet	AssurAnalytics — Analyse des Sinistres & Profil des Assurés
Stack	Python · Dash · Plotly · Pandas · Bootstrap
Année	2026

13.

10.2 Valeur GitHub & CV

Ce projet démontre les compétences suivantes, valorisables sur LinkedIn et GitHub :

- Data Engineering — chargement, nettoyage, enrichissement de données avec Pandas
- Data Visualization — 13 graphiques Plotly couvrant tous les types d'insights visuels
- Python Web — framework Dash avec callbacks réactifs et architecture modulaire
- UX/UI — design professionnel Bootstrap, CSS avancé, expérience utilisateur soignée
- Business Intelligence — storytelling automatique, KPIs, recommandations tarifaires
- Export multi-format — génération Excel, HTML, PDF depuis une interface web
- Best Practices — structure modulaire app/layout/callbacks, code commenté et lisible

❖ Conseil pour les recruteurs : ce projet illustre une capacité à transformer un jeu de données métier en outil d'aide à la décision opérationnel, de bout en bout, de la donnée brute à l'export du rapport.