

Université de Reims Champagne-Ardenne

M2 Statistique pour l'évaluation et la prévision

Évaluation du coût environnemental d'un vêtement

Membres de l'équipe

Christelle KOUDORO ; Bastien DEMARIN ; Ruxandra ILIESCU ;
Nicolas YACKOB ; Killian NICOLAS

Décembre, 2024

Table des matières

| | |
|---|----|
| Liste des figures | 2 |
| Liste des tableaux..... | 2 |
| 1. Contexte et objectifs du projet..... | 3 |
| 1.1. Contexte | 3 |
| 1.2. Objectifs du projet..... | 4 |
| 2. Fonctionnement de l'application..... | 5 |
| 2.1. Présentation de l'application | 5 |
| 2.2. Gestion des erreurs dans l'application | 9 |
| 2.3. Interaction entre VBA et Python..... | 10 |
| 3. Méthodologie | 11 |
| 3.1. Variable d'intérêt | 11 |
| 3.2. Variables explicatives | 12 |
| 3.3. Modèle de prédiction..... | 13 |
| 3.5. Sources des données..... | 14 |
| 4. Organisation du travail et difficultés..... | 18 |
| 4.1. Organisation du travail | 18 |
| 4.2. Difficultés rencontrées et perspectives..... | 20 |
| Bibliographie..... | 22 |
| Annexes..... | 23 |

Liste des figures

| | |
|---|----|
| Figure 1 : Page d'accueil de l'application | 5 |
| Figure 2 : Page 1- sélection..... | 6 |
| Figure 3 : Page 2-Formulaire | 7 |
| Figure 4 : Page 3- Sortie des résultats..... | 8 |
| Figure 5 : Distribution des impacts environnementaux en fonction du type de produit..... | 17 |
| Figure 6 : Distribution des impacts environnementaux en fonction du type de matériel | 18 |

Liste des tableaux

| | |
|---|----|
| Tableau 1 : Catégories de produit et leur poids | 14 |
| Tableau 2 : Les matériaux | 14 |
| Tableau 3 : Le pays de provenance | 15 |
| Tableau 4 : Répartition des impacts par catégorie de produit | 16 |

1. Contexte et objectifs du projet

1.1. Contexte

La mode, bien qu'indispensable, est l'une des industries les plus polluantes au monde, contribuant significativement aux émissions de gaz à effet de serre, à la consommation d'eau et à la génération de déchets textiles. Avec 4 milliards de tonnes de CO₂ émises annuellement, soit 10 % des émissions mondiales de gaz à effet de serre (GES), l'industrie textile contribue de façon significative au réchauffement climatique. Selon l'Agence européenne pour l'environnement, les achats de textile dans l'UE en 2020 ont généré des émissions de CO₂ équivalentes à 270 kg par personne. Cela signifie que les produits textiles consommés dans l'UE ont généré des émissions de gaz à effet de serre de 121 millions de tonnes.

L'industrie textile fait partie des plus polluantes au monde. Elle rejette 240 000 tonnes de microfibres plastiques chaque année ; 20 % de la pollution des eaux est attribuable à la teinture et au traitement des vêtements et il faut 7 500 litres d'eau pour fabriquer un jean en coton. La production textile est très gourmande en eau, et exige en outre des terres pour cultiver le coton et d'autres fibres. Selon un article de vie publique, au niveau européen, l'habillement est la quatrième catégorie de consommation ayant le plus d'impact sur l'environnement, après l'alimentation, le logement et les transports. En effet, le rythme de croissance de la demande d'habillement s'intensifie avec une célérité inégalée : de 62 millions de tonnes en 2015 à 102 millions de tonnes en 2030 d'après un rapport du WWF sur l'industrie de l'habillement et des textiles. D'après l'Agence Européenne pour l'Environnement (2020), la production mondiale de fibres textiles a presque doublé. De 58 millions de tonnes en 2000 à 109 millions de tonnes en 2020, elle devrait atteindre 145 millions de tonnes d'ici 2030.

En dépit de cette forte demande, 80% des consommateurs français affirment vouloir prendre en compte l'impact environnemental de leurs achats, mais qu'ils disposent généralement de peu d'informations fiables pour agir efficacement selon l'enquête Ipsos MORI réalisée en 2019. Face à une prise de conscience croissante des consommateurs, les outils permettant d'évaluer l'impact environnemental des produits de consommation, comme les vêtements, deviennent essentiels pour encourager des choix responsables.

C'est dans ce cadre que le 3 avril 2024, le ministère de la Transition écologique et de la Cohésion des territoires a dévoilé « Ecobalyse », un nouvel outil pour calculer le coût environnemental des produits textiles. Cette démarche fait suite aux propositions de la

Convention Citoyenne pour le Climat, et a vocation à appliquer la loi Climat et Résilience du 24 août 2021. Développé avec l'ADEME (Agence de la transition écologique), Ecobalyse est destiné aux professionnels du secteur. Sauf qu'étant destiné aux professionnels, les informations demandées aux utilisateurs sont très pointues et ne permettent pas à un simple consommateur d'évaluer le coût environnemental de son vêtement.

C'est dans ce cadre que vient prendre place notre application qui est plus destinée plus consommateurs. Il leur sera demandé des informations moins pointues et faciles d'accès sur les étiquettes de leurs vêtements. Elle est plus intuitive et plus facile à prendre en main par les utilisateurs.

1.2. Objectifs du projet

L'objectif de notre application est de pouvoir aider toute personne qui le souhaite à pouvoir évaluer le coût environnemental de son vêtement. Grâce à notre interface, l'utilisateur entrera des informations qui lui seront demandées sur le vêtement dont il souhaite évaluer le coût environnemental. Il aura en réponse, une estimation et une interprétation du score du coût environnemental du vêtement ainsi que des propositions pour des choix plus écologiques. Ainsi nous souhaitons proposer un outil digital et ergonomique qui :

- estime l'impact environnemental des vêtements via un score
- s'appuie sur des informations dont l'utilisateur lambda disposera sur l'article vestimentaire
- est facile à prendre en main et à utiliser par tout utilisateur.

2. Fonctionnement de l'application

2.1. Présentation de l'application

La page d'accueil de notre application se présente comme suit :

Figure 1 : Page d'accueil de l'application



Source : Les auteurs, 2024

Sur cette page, nous expliquons l'intérêt de notre application ainsi que les réponses qu'elle peut apporter pour répondre aux problématiques des potentiels utilisateurs. Alors l'utilisateur pourra accéder à notre analyse sur le coût environnemental lié à la production des vêtements en appuyant sur le bouton 'Formulaire'. Après avoir actionné ce bouton, l'utilisateur sera redirigé vers la première page du questionnaire.

Figure 2 : Page 1- sélection

APPLICATION

Sélectionnez et analysez
votre vêtement en cliquant
sur l'image le représentant

Haut du corps



Bas du corps



Divers



Quitter la page

Source : Les auteurs, 2024

Une fois sur cette page, il doit cliquer dans un premier temps sur l'un des boutons entre 'Haut du corps', 'Bas du corps' et 'Divers' afin de sélectionner le type de vêtement dont il souhaite évaluer le coût environnemental. Suite à son choix, un formulaire comportant des questions auxquelles il doit répondre sur son vêtement lui est présenté. Le formulaire se présente comme sur la figure 3.

Figure 3 : Page 2-Formulaire

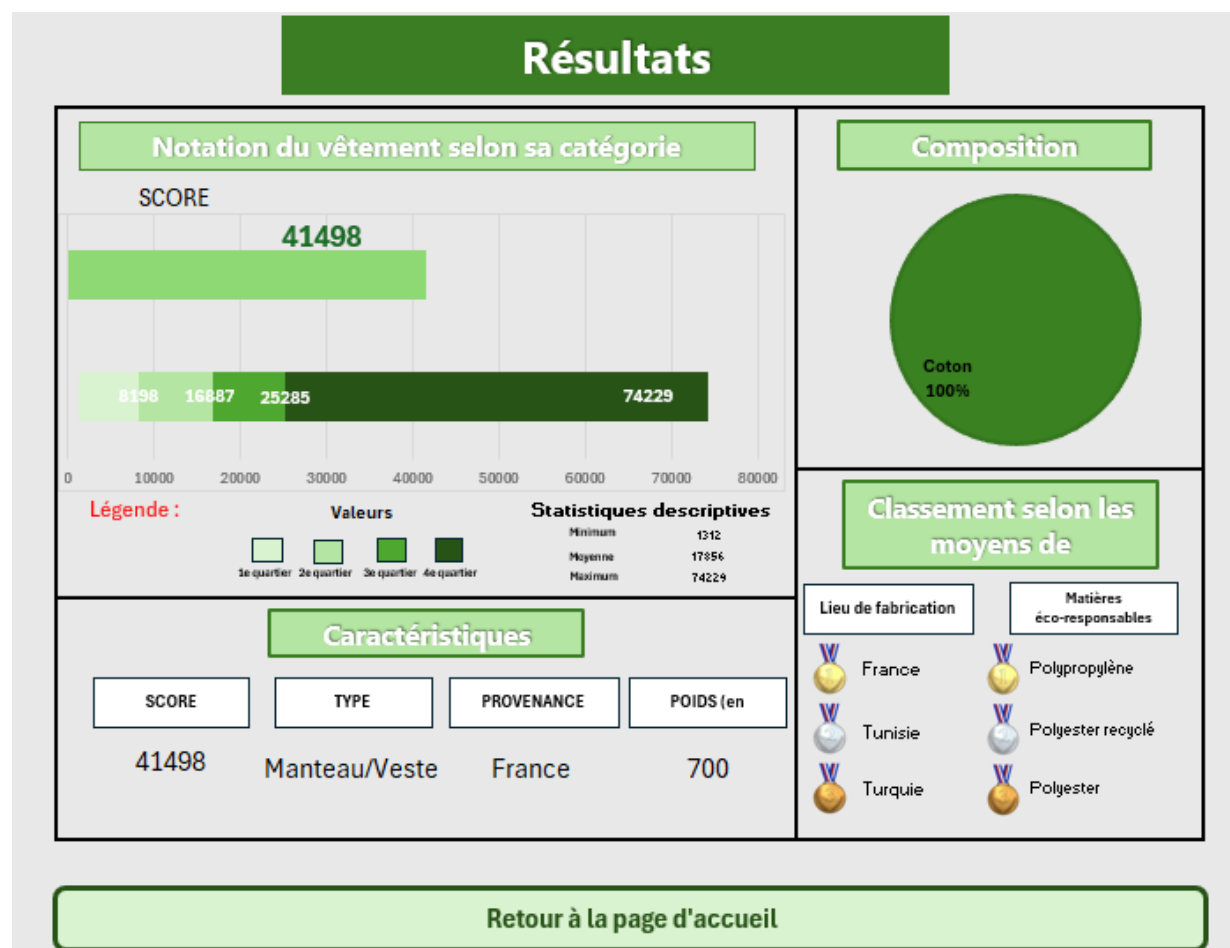
| Caractéristiques du vêtement | |
|--|---|
| Catégorie du vêtement : <input type="text"/> | |
| Poids du vêtement (en gramme) : <input type="text"/> | Provenance (Made in) : <input type="text"/> |
| Matière(s) première(s) : <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> | Pourcentage de la matière : <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> |
| Retour à l'accueil | Réinitialisation Analyse |

Source : Les auteurs, 2024

Il lui est demandé des informations sur la catégorie, le poids, la provenance puis la composition du vêtement. Pour toutes les questions exceptées celle concernant le poids, l'utilisateur a une liste déroulante afin de pouvoir faire un choix. Les modalités de réponses des différentes questions se basent sur la configuration de nos données. Nous avons opté pour ce choix afin de limiter les erreurs de saisie. Le poids demandé doit être donné en grammes. Après avoir rempli ce formulaire, l'utilisateur clique ensuite sur le bouton 'Analyse' qui lui renvoie la page ci-dessous comportant les résultats en quatre parties. Dans la première nommée « notation vêtement selon sa catégorie », il a une interprétation du score. L'interprétation consiste dans un premier temps au positionnement du score de son vêtement sur une barre parallèlement à une seconde barre qui présente les score des vêtement de la même catégorie que le sien présents dans notre base. Cela est fait de de sorte à ce qu'il puisse facilement comparer son vêtement à

ceux de la même catégorie afin de lui permettre de se situer. En fonction du positionnement il peut voir clairement si son score est plutôt élevé ou pas avec la prononciation de la couleur sur la barre du bas. Il a également une légende et les statistiques descriptives lui donnant le minimum, la moyenne et le maximum des score des vêtements de la même catégorie. Dans les parties « Composition » et « Caractéristiques », l'utilisateur retrouve les informations qu'il a saisies précédemment, soient le type, la provenance, le poids et la composition en plus du score de son vêtement. Dans la dernière nommée « Classement », on lui fait une proposition des meilleurs matériaux éco-responsables et des meilleurs lieux de provenance qu'il devrait privilégier afin d'avoir le même vêtement mais avec un moindre score environnemental.

Figure 4 : Page 3- Sortie des résultats



Source : Les auteurs, 2024

Après la lecture des résultats, l'utilisateur peut cliquer sur le bouton 'Retourner à la page d'accueil' pour retourner sur la première page de l'application.

2.2. Gestion des erreurs dans l'application

Dans le formulaire, nous avons implémenté des restrictions afin de contrôler les réponses saisies par l'utilisateur pour chaque variable à renseigner. Nous allons vous présenter nos restrictions concernant le poids, la provenance, les matières et leurs pourcentages.

- **Poids** : Le poids du vêtement est renseigné dans une zone de texte (Textbox) et est exprimé en gramme, l'utilisateur est libre d'inscrire n'importe quelle valeur numérique comprise entre 1 et 4000 grammes. Cette plage a été définie pour éviter les valeurs extrêmes qui ne correspondent pas à un poids réaliste pour un vêtement. De plus, nous avons mis des interdictions pour que l'utilisateur ne puisse pas inscrire de valeurs catégorielles.
- **Provenance (Made in)** : La provenance est sélectionnée via une liste déroulante où les pays et les régions sont déjà enregistrés, avec la base de données que nous avons récoltées, alors l'utilisateur n'a pas le choix lors de la sélection et ne peut sélectionner que ces valeurs-ci. Il n'y a donc pas de possibles erreurs sur cette variable.
- **Matières** : Une liste déroulante avec tous les matériaux pouvant composer un vêtement. De la même façon que la provenance, l'utilisateur ne peut pas émettre des valeurs. Pour chaque type de matière, son pourcentage doit être renseigné si soit la matière 1 ou soit le pourcentage de la matière 1 est renseigné, mais pas l'autre variable alors un message d'erreur sera envoyé pour prévenir que ces informations sont inexactes ou mal renseignées, ce système est fait sur l'ensemble des matières inscrites.
- **Pourcentage des matières** : TextBox où l'utilisateur peut inscrire les valeurs correspondant au pourcentage de la matière. Dans chaque texte, si on a une valeur catégorielle alors la réponse est supprimée et un message d'erreur. Ensuite les 4 textbox sont connectés ensemble, c'est-à-dire que pour que le total des pourcentages des matières doit être compris entre 70 et 100 pour avoir un score réaliste. Si l'ensemble des valeurs est supérieur ou inférieur à ces échelles alors un message d'erreur est envoyé pour prévenir du problème.

Si toutes ces informations sont correctement remplies selon les restrictions imposées, la page des résultats s'affichera.

2.3. Interaction entre VBA et Python

Faire communiquer VBA avec un script Python était l'un des enjeux majeurs de ce projet. Rappelons que notre objectif est de déterminer le coût environnemental d'un vêtement à partir. Nous avons décidé que l'utilisateur devait sélectionner, pour chaque caractéristique, une valeur parmi une liste proposée dans un formulaire VBA. Cela nous permet, grâce à des mécanismes de sécurité intégrés au formulaire, de nous assurer que les données saisies sont conformes aux attentes pour leur traitement par le script Python. Plusieurs options s'offraient à nous pour transmettre ces données au script Python.

- **Utilisation d'un fichier CSV comme intermédiaire**

Cette méthode consiste à écrire les données saisies dans le formulaire dans un fichier CSV, puis à lancer un script Python pour lire ce fichier et en extraire les informations. L'avantage de cette approche est qu'elle permet de conserver un historique des données saisies par l'utilisateur.

- **Transmission directe via les arguments du script**

Une autre option était de transmettre les données saisies dans le formulaire directement en tant que arguments du script Python, où elles seraient décodées. Cette méthode est plus directe que la précédente puisqu'elle supprime l'étape intermédiaire de gestion d'un fichier CSV.

- **Utilisation d'une API**

Nous avons également envisagé la possibilité de recourir à une API pour transmettre les données. Bien que nous n'ayons pas approfondi cette solution dans le cadre du projet actuel, elle pourrait constituer une évolution future pour notre application.

Nous avons retenu la deuxième option, et son fonctionnement sera détaillé dans les sections suivantes.

1. Fonctionnement de la transmission directe

Dans un premier temps, l'utilisateur sélectionne des valeurs dans les différents champs du formulaire VBA. Si tous les champs nécessaires pour effectuer la prédiction ne sont pas correctement remplis, la validation est bloquée. Cela nous permet de garantir un contrôle total sur la structure des données envoyées au script Python.

Les données sont transmises sous la forme d'une chaîne de caractères structurée, par exemple : T-shirt/Polo S Chine Chanvre 30 Coton 70. La structure de cette chaîne reste constante :

`type_de_vêtement taille pays_de_fabrication matériau_1 %_matériau_1 ... matériau_n %_matériau_n`.

Cette organisation simplifie l'extraction des variables dans le script Python avec chaque information occupant une position fixe. Par exemple, le type de vêtement est toujours le premier élément de la chaîne.

2. Gestion des matériaux multiples

La seule difficulté résidait dans la gestion des listes de matériaux, qui peuvent varier en nombre. Initialement, nous avons limité l'application à des vêtements composés d'un seul matériau, ce qui simplifiait le développement des autres fonctionnalités. Cependant, nous avons depuis étendu la prise en charge aux vêtements composés de plusieurs matériaux.

Pour cela, nous exploitons la structure de la chaîne reçue. Une boucle est utilisée pour organiser les matériaux et leurs pourcentages dans deux listes distinctes. Bien que des solutions plus complexes aient pu être envisagées, nous avons opté pour cette approche simple en raison du faible volume de données à traiter et de notre contrôle total sur leur structure. Cette méthode nous a permis d'assurer une interaction efficace et fiable entre VBA et Python tout en garantissant la flexibilité nécessaire à l'évolution future de l'application.

3. Méthodologie

3.1. Variable d'intérêt

La variable d'intérêt dans notre modèle est le **score du coût environnemental total** d'un vêtement, noté **Y**. Ce coût est déterminé par deux principales composantes que seront le coût de fabrication et le coût de transport. Il est représenté par la formule suivante :

$$Y \sim \text{coût_fabrication} + \text{coût_transport}$$

avec :

- **Y** : Score du coût environnemental total
- **coût_fabrication** : Coût environnemental lié à la fabrication du vêtement
- **coût_transport** : Coût environnemental lié au transport du vêtement

3.2. Variables explicatives

1. Coût de fabrication

Le coût de fabrication d'un vêtement est calculé à partir de trois variables principales :

$\text{coût_fabrication} \sim \text{poids_vêtement} + \text{composition} + \text{made_in}$

avec :

- **poids_vêtement** : Le poids du vêtement, influencé par sa taille, son type et sa composition.
- **composition** : Le type de matériau utilisé pour fabriquer le vêtement (ex. : coton, polyester).
- **made_in** : Le lieu de fabrication du vêtement, qui peut avoir une influence sur l'impact environnemental dû aux réglementations locales et aux pratiques industrielles.

2. Poids du vêtement

Le poids du vêtement est une variable essentielle dans le calcul du coût de fabrication, néanmoins, il est difficile à obtenir pour l'utilisateur. Nous avons décidé de le modéliser comme suit :

$\text{poids_vêtement} \sim \text{taille_vêtement} + \text{type_vêtement} + \text{composition}$

avec :

- **taille_vêtement** : La taille du vêtement (par exemple S, M, L), qui influe sur la quantité de tissu nécessaire.
- **type_vêtement** : Le type de vêtement (par exemple t-shirt, pantalon, veste), qui affecte aussi la quantité de tissu et la complexité de la production.
- **composition** : Le matériau utilisé pour le vêtement, qui impacte à la fois le poids et le coût environnemental.

Dans un premier temps, nous voulions permettre à l'utilisateur de ne pas avoir à le saisir en utilisant plutôt la taille du vêtement et sa composition pour estimer le poids. Cette approche aurait rendu l'application moins contraignante à utiliser.

Pour cela, nous avons envisagé deux approches. La première consistait à scraper le site de Décathlon. Sur ce dernier, on peut, pour certains articles, retrouver la masse, la composition et

la taille du vêtement. Cependant, il existe de grandes irrégularités au sein des pages des articles, ce qui rendait le scraping complexe à réaliser. De plus, même pour les articles pour lesquels les informations étaient correctement agencées dans la page, nous n'arrivions pas à obtenir la bonne valeur. Faute de temps et de moyens, nous avons donc abandonné cette approche.

Notre seconde approche aurait été de calculer le poids en fonction de la surface de tissu/matériau utilisée, multipliée par la masse volumique du matériau. Encore une fois, nous avons rencontré des difficultés. Il n'existe pas de base de données référençant la surface de tissu utilisée pour la confection des vêtements, en dehors des tailleurs.

De plus, ces deux approches ne tenaient pas compte de la disparité de poids entre plusieurs vêtements d'un même type. Par exemple, deux t-shirts d'une même taille et de proportions identiques peuvent avoir un poids différent à cause du grammage du tissu.

Nous avons donc fait une concession et réintroduit le poids du vêtement parmi les variables demandées dans le formulaire.

3. Coût de transport

Le coût de transport est directement influencé par deux facteurs :

$\text{coût_transport} \sim \text{made_in} + \text{poids_vêtement}$

Le type de transport n'étant pas connu, il sera négligé ici mais sera représenté implicitement dans par la variable `made_in`.

- **made_in** : Le lieu de fabrication, qui détermine la distance de transport entre le site de production et le lieu de vente ou de consommation.
- **poids_vêtement** : Le poids du vêtement, qui influence les coûts de transport, car un vêtement plus lourd nécessitera plus d'énergie pour être transporté.

3.3. Modèle de prédiction

Nous avons testé plusieurs types de modèles. Le modèle le plus concluant est le modèle Gradient Boosting Machine (GBM).

Le GBM construit des modèles en se concentrant progressivement sur les erreurs des arbres précédents, ce qui peut permettre de mieux s'adapter aux non-linéarités et aux interactions complexes entre les variables.

3.5. Sources des données

Certaines des variables ont été récupérées via des APIs et d'autres directement à partir des informations fournies par l'utilisateur :

- **poids du vêtement, composition** et **made_in** sont des informations que l'utilisateur peut fournir directement.
- **coût de fabrication** et **coût de transport** ont été obtenus via l'API d'Ecobalyse dans un premier temps, puis par des modèles de prédiction.
- Enfin, nous avons exploité notre **modèle global** pour obtenir **Y**, c'est-à-dire le **coût environnemental total** d'un vêtement.

Pour entraîner notre modèle, nous avons scrappé les données nécessaires via l'API Ecobalyse pour mettre en place une base de données assez exhaustive, en faisant varier nos variables comme suit :

Tableau 1 : Catégories de produit et leur poids

| Catégorie de produit | Intervalle des poids prises en compte (kg) |
|----------------------|--|
| Caleçon | 0,02 - 0,2 |
| Chaussettes | 0,005 - 0,2 |
| Chemise | 0,05 - 0,5 |
| Jean | 0,25 - 1,5 |
| Jupe | 0,05 - 1 |
| Maillot de bain | 0,05 - 0,4 |
| Manteau | 0,2 - 4 |
| Pantalon | 0,1 - 1,5 |
| Pull | 0,1 - 1,5 |
| T-shirt | 0,05 - 0,5 |
| Slip | 0,01 - 0,15 |

Source : Les auteurs, 2024

Tableau 2 : Les matériaux

| Matériel | Matière première | Origine | Recyclé ? |
|------------|---|--------------------------------------|-----------|
| coton-rdp | Coton recyclé (déchets de production) | Matière naturelle d'origine végétale | Oui |
| coton-rdpc | Coton recyclé (déchets post-consommation) | Matière naturelle d'origine végétale | Oui |

| | | | |
|-------------------------------|---------------------------|---|-----|
| ei-acrylique | Acrylique | Matière synthétique | Non |
| ei-chanvre | Chanvre | Matière naturelle d'origine végétale | N/A |
| ei-coton | Coton | Matière naturelle d'origine végétale | Non |
| ei-coton-organic | Coton biologique | Matière naturelle d'origine végétale | Non |
| ei-jute-kenaf | Jute | Matière naturelle d'origine végétale | Non |
| ei-laine-nouvelle- filiere | Laine nouvelle filière | Matière naturelle d'origine animale | Non |
| ei-laine-par-defaut | Laine par défaut | Matière naturelle d'origine animale | Non |
| ei-lin | Lin | Matière naturelle d'origine végétale | Non |
| ei-pa | Nylon | Matière synthétique | Non |
| ei-pet | Polyester | Matière synthétique | Non |
| ei-pet-r | Polyester recyclé | Matière synthétique | Oui |
| ei-pp | Polypropylène | Matière synthétique | Non |
| ei-viscose | Viscose | Matière artificielle d'origine organique | Non |
| elasthane | Elasthanne (Lycra) | Matière synthétique | Non |

Source : Les auteurs, 2024

Tableau 3 : Le pays de provenance

| Abréviation | Pays |
|-------------|----------------------------|
| --- | Pays inconnu |
| BD | Bangladesh |
| TR | Turquie |
| TN | Tunisie |
| ROC | Région - Océanie |
| RNA | Région - Amérique du nord |
| RME | Région - Moyen-Orient |
| RLA | Région - Amérique Latine |
| REO | Région - Europe de l'Ouest |
| REE | Région - Europe de l'Est |
| RAS | Région - Asie |
| RAF | Région - Afrique |

| | |
|----|----------|
| PK | Pakistan |
| MM | Myanmar |
| MA | Maroc |
| KH | Cambodge |
| IN | Inde |
| FR | France |
| CN | Chine |
| VN | Vietnam |

Source : Les auteurs, 2024

La variable `impact_ecs`, donne l'impact environnemental, exprimé en points d'impact. Un point d'impact correspond à l'impact environnemental annuel d'un habitant européen.

Pour chaque produit, nous avons récupéré les impacts ecs de 10 poids différents équitablement répartis sur l'intervalle indiqué. Pour chaque poids, nous avons récupéré les impacts ecs suivant chaque pays. Finalement, pour chaque pays nous avons récupéré les impacts ecs suivant chaque composition de produit.

Au total, il y a 35200 enregistrements dans la base de données :

- 3200 enregistrements pour chacune des 11 catégories de produit
- 2200 enregistrements pour chacune des 16 catégories de matériel
- 1760 enregistrements pour chacun des 20 pays de provenance

On note une corrélation assez élevée entre l'impact environnemental d'un produit et son poids (0.96), fait qui va justifier notre choix de modèle pour la suite.

Par catégorie de produit, on a les impacts répartis comme suit, exprimés en points d'impact environnemental :

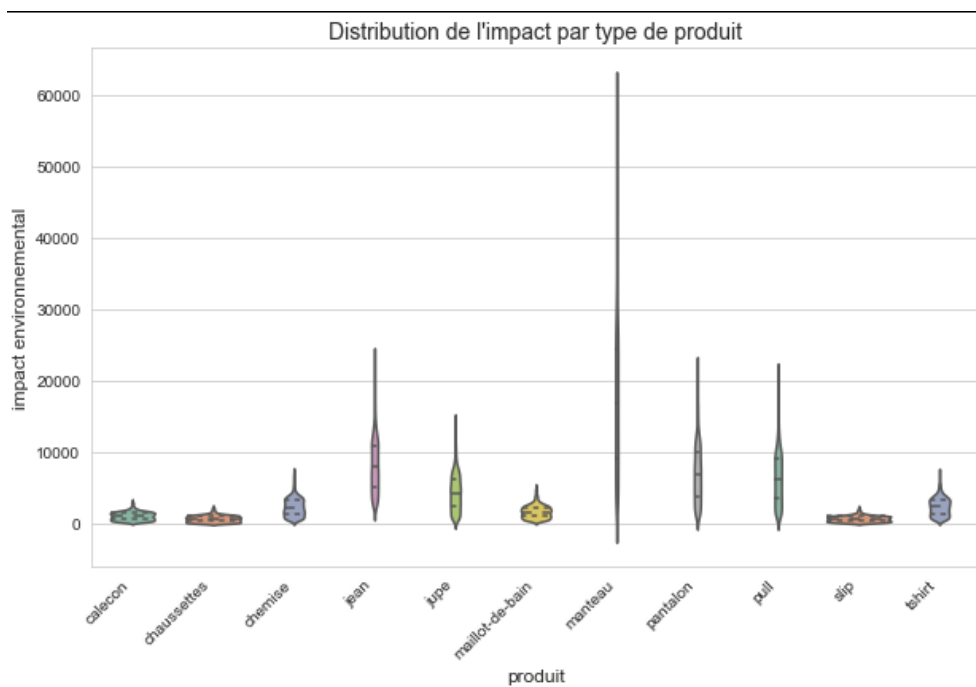
Tableau 4 : Répartition des impacts par catégorie de produit

| Produit | Impact Min | Impact Q1 | Impact Moyen | Impact Q3 | Impact Max |
|-------------|------------|-----------|--------------|-----------|------------|
| caleçon | 172 | 599 | 1141 | 1577 | 4064 |
| chaussettes | 34 | 349 | 778 | 1125 | 3163 |
| chemise | 366 | 1270 | 2474 | 3392 | 9390 |

| | | | | | |
|-----------------|------|------|--------|--------|--------|
| jean | 1871 | 4848 | 8399 | 11 096 | 29 686 |
| jupe | 351 | 2123 | 4586 | 6513 | 18 282 |
| maillot de bain | 309 | 963 | 1801 | 2435 | 6778 |
| manteau | 1313 | 8198 | 17 857 | 25 286 | 74 229 |
| pantalon | 745 | 3518 | 7316 | 10 292 | 28 625 |
| pull | 631 | 3171 | 6709 | 9395 | 27 587 |
| slip | 86 | 391 | 776 | 1087 | 2933 |
| t shirt | 380 | 1302 | 2520 | 3463 | 9500 |

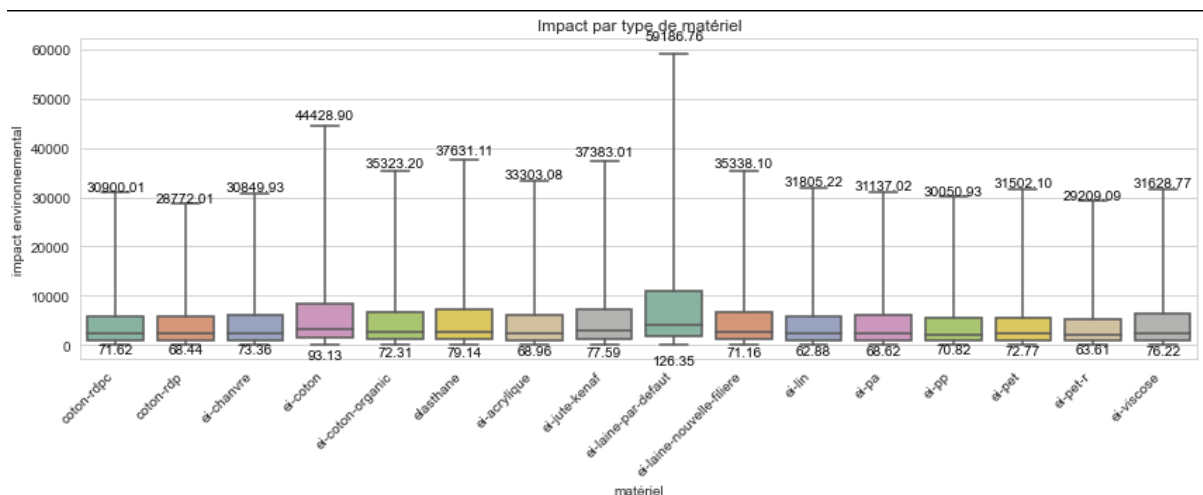
Source : Les auteurs, 2024

Figure 5 : Distribution des impacts environnementaux en fonction du type de produit



Source : Les auteurs, 2024

Figure 6 : Distribution des impacts environnementaux en fonction du type de matériel



Source : Les auteurs, 2024

Nous vérifions s'il y a une différence significative entre les impacts par type de produit par un test de Kruskal-Wallis. Nous constatons que c'est le cas et par un test post-hoc de Dunn on trouve qu'on peut parler de significativité entre tous les groupes deux à deux.

4. Organisation du travail et difficultés

4.1. Organisation du travail

Pour la réalisation de ce projet, nous avons d'abord établi une roadmap, qui a été mise à jour progressivement en fonction des réalités rencontrées au fil de l'avancement. Cette roadmap a ensuite été déclinée en sprints, chacun associé à des objectifs précis et des e Definition of Done clairement définis pour mesurer l'atteinte des résultats. Les différents sprints se présentent comme suit :

Sprint 1 : 09 oct- 23oct 2024

- **Objectifs :**
 - Concevoir un produit analytics en réponse à un besoin client identifié.
 - Construire la roadmap produit découpant les différentes étapes en différents lots de valeur.
 - Construire un premier jeu de données permettant de traiter la problématique.
 - Valider la prise en main du setup de développeur.

- **Définitions of Done :**

- Des maquettes d’interface sont construites et permettent de comprendre le parcours de l’utilisateur au sein de l’application.
- Une roadmap produit détaillée par compétences est construite et découpée dans le temps en différents lots de valeur.
- La variable d’intérêt et les variables explicatives sont présentées de façon explicite & documentées au sein du rapport.
- Un premier script python exécutable est disponible au sein du projet github validant l’autonomie de l’équipe dans l’utilisation des outils de développements.

Sprint 2 : 24 oct - 08 nov 2024

- **Objectifs :**

- Concevoir une interface qui donne le score environnemental d’un vêtement unique.
- Valider le modèle d’estimation du score environnemental d’un vêtement unique.
- Construire un premier jeu de données permettant de traiter la problématique.

- **Définitions of Done :**

- Une interface fonctionnelle demandant à l’utilisateur les données du vêtement et lui retournant le score environnemental est construite.
- Un script Python comportant le modèle de prédiction retournant le score du vêtement à l’utilisateur est disponible sur github.
- Une base de données contenant les variables poids, composition et lieu de fabrication est construite avec documentation du processus détaillé.

Sprint 3 : 09 nov - 21 nov 2024

- **Objectifs de sprint :**

- Concevoir une interface qui donne le score et l’interprétation associé.
- Améliorer le modèle d’estimation du score environnemental d’un vêtement.
- Documenter le fonctionnement de l’application.

- **Définitions of Done :**

- Une interface retournant à l’utilisateur le score de son vêtement avec une interprétation est conçue.

- Un script Python comportant le modèle de prédiction du score d'un vêtement composé de plusieurs matériaux est disponible sur github.
- Un chapitre indiquant le fonctionnement de l'application est rédigé dans le rapport.

Sprint 4 : 22 nov – 09 déc 2024

- **Objectifs de sprint :**
 - Améliorer l'interprétation du score.
 - Renforcer le pouvoir prédictif du modèle.
- **Définitions of Done :**
 - Une interface retournant à l'utilisateur une interprétation du score de son vêtement avec un graphe est conçue.
 - Un script Python comportant le modèle de prédiction avec une erreur de moins de 7,69% du score d'un vêtement est disponible sur github.

Sprint 5 : 10 déc- 19 déc 2024

- **Objectifs de sprint :**
 - Améliorer l'ergonomie de l'interface des résultats.
 - Renforcer le pouvoir prédictif du modèle.
- **Définitions of Done :**
 - Une interface des résultats claire, intuitive et facilement interprétable par l'utilisateur, avec une lisibilité améliorée, est conçue.
 - Un script Python intégrant le modèle de prédiction, ainsi que le calcul du malus pour le score d'un vêtement composé de plusieurs matériaux, est disponible sur GitHub.

4.2. Difficultés rencontrées et perspectives

Nos principales difficultés ont été liées à la gestion des données. Étant donné qu'aucun travail préalable n'existait dans ce domaine, nous n'avons pas pu accéder à une base de données déjà disponible pour mener à bien notre projet. Par conséquent, nous avons dû recourir au scraping de données via l'API Ecobalyse. Cette méthode, bien qu'essentielle pour obtenir les informations nécessaires, s'est révélée chronophage et a mobilisé une part importante de nos ressources temporelles.

Malheureusement, les délais impartis pour la remise du produit final et les difficultés rencontrées lors des tentatives de scraping sur le site de Decathlon nous ont empêchés d’avoir les données nécessaires pour estimer le poids des vêtements de manière automatisée. En conséquence, nous avons dû inclure dans notre formulaire une question demandant directement à l’utilisateur de renseigner le poids de son vêtement. Nous sommes conscients que cette démarche complique l’expérience utilisateur et pourrait constituer un frein à l’adoption de notre application.

Pour une éventuelle reprise du projet à l’avenir, nous recommandons de prioriser la collecte de données réelles permettant d’estimer automatiquement le poids des vêtements en fonction de leur taille et de leur composition. Cette approche non seulement simplifierait l’utilisation de l’application pour l’utilisateur, mais améliorerait également la fiabilité et l’autonomie du système.

Bibliographie

1. <https://ecobalyse.beta.gouv.fr/#/textile/simulator>
2. <https://www.notre-environnement.gouv.fr/actualites/breves/article/ecobalyse-calculez-le-cout-environnemental-de-vos-vetements#>
3. <https://www.europarl.europa.eu/topics/fr/article/20201208STO93327/production-et-dechets-textiles-les-impacts-sur-l-environnement-infographies>

Annexes

Annexe 1 : Roadmap produit



Annexe 2 : Description des données

Plusieurs jeux de données sont utilisés lors du fonctionnement de notre application. Ils servent principalement à établir des correspondances entre les données saisies par l'utilisateur dans le formulaire VBA et les variables utilisées pour l'entraînement du modèle de prédiction. En voici une description.

- countries.csv :

Cette base de données est une table de correspondance qui associe des codes géographiques à leurs noms respectifs, permettant d'identifier des pays ou des régions. La colonne code contient des abréviations ou des identifiants, tandis que la colonne name fournit les noms correspondants. Elle inclut des régions globales représentées par des codes commençant par "R", comme REO pour "Région - Europe de l'Ouest" ou RLA pour "Région - Amérique Latine", ainsi que des pays identifiés par des codes internationaux à deux lettres conformes à la norme ISO 3166-1 alpha-2, tels que FR pour "France" ou CN pour "Chine". Le code spécial --- désigne un "Pays inconnu" et sert de valeur par défaut pour les données non spécifiées.

- materials.csv :

Cette base de données est une table de correspondance entre des matériaux et leurs identifiants uniques (codes). La colonne id contient les codes ou abréviations qui servent à identifier chaque matériau dans un système, tandis que la colonne name fournit le nom complet et parfois une description détaillée. Les matériaux référencés incluent des fibres naturelles comme le coton (ei-coton), le chanvre (ei-chanvre), et le lin (ei-lin), ainsi que des matériaux recyclés comme le coton recyclé (coton-rdpc et coton-rdp) et le polyester recyclé (ei-pet-r).

- products.csv :

Cette base de données est une table de correspondance qui associe des identifiants (id) à des catégories de vêtements (name). La colonne id contient des codes uniques permettant d'identifier chaque type de vêtement, tandis que la colonne name fournit une description textuelle complète du vêtement ou d'une catégorie associée. Parmi les articles référencés, on trouve des vêtements spécifiques comme la chemise (chemise), le jean (jean), ou le pull (pull). Certaines catégories couvrent plusieurs articles proches, comme jupe pour "Jupe/Robe" ou pantalon pour "Pantalon/Short". D'autres articles incluent des sous-vêtements, tels que le caleçon (calecon) et le slip/boxer (slip), ainsi que des vêtements spécifiques comme le maillot de bain (maillot-de-bain) et des accessoires comme les chaussettes (chaussettes).

- size_to_weight.csv :

Cette base de données est une table reliant les types de vêtements à leur poids approximatif, en grammes, pour différentes tailles standardisées (XS, S, M, L, XL). La colonne Type identifie le type de vêtement, tandis que les colonnes suivantes fournissent le poids associé à chaque taille.

Les poids varient significativement en fonction du type de vêtement. Par exemple, une chemise pèse entre 175 g pour la taille XS et 375 g pour la taille XL, tandis qu'un manteau est beaucoup plus lourd, allant de 800 g (XS) à 1500 g (XL). Des articles plus légers comme un T-shirt pèsent entre 115 g (XS) et 225 g (XL), et des accessoires tels que les chaussettes ne pèsent que 25 g (XS) à 65 g (XL). Les sous-vêtements, comme le caleçon (65 g à 140 g) et le slip/boxer (60 g à 140 g), ont des poids similaires. Le jean, avec un poids allant de 400 g (XS) à 750 g (XL), illustre également une augmentation substantielle avec la taille.

Annexe 3 : Description des scripts et des fonctions

- main.py

Ce programme sert d'entrée à l'application. Il est composé d'une unique fonction. Cette dernière prend comme argument une chaîne de caractère et lance la fonction trigger du module steps.

- steps.py

Ce module sert à estimer l'impact environnemental d'un vêtement en fonction des informations fournies par un utilisateur via un formulaire VBA. Il automatise le traitement des données utilisateur, les traduit en variables adaptées à un modèle de prédiction, et utilise un algorithme d'apprentissage automatique pour fournir une estimation quantitative de cet impact.

1. trigger (user_clothing_data: str)

La fonction `trigger` orchestre l'ensemble des étapes nécessaires pour prédire l'impact environnemental d'un vêtement à partir des données saisies par l'utilisateur dans un formulaire VBA. Elle effectue successivement les étapes suivantes : extraction et structuration des données d'entrée (via `step_1`), conversion des données en variables explicatives adaptées au modèle prédictif (via `step_2`), construction des jeux de données explicatifs pour chaque matériau (via `step_3`), et enfin prédiction de l'impact environnemental global en combinant les résultats partiels (via `step_4`). Cette fonction est l'entrée principale du module et fournit un flux de traitement automatisé et cohérent des données utilisateur.

2. step_1(user_clothing_data: str)

La fonction `step_1` extrait et structure les informations fournies par l'utilisateur dans le formulaire VBA. Ces informations incluent le nom du produit, le pays d'origine, la taille, ainsi que la composition des matériaux (types et pourcentages). En utilisant une fonction utilitaire pour diviser la chaîne d'entrée en mots, elle identifie les éléments et les organise en un format structuré. Les matériaux et leurs pourcentages sont traités séparément, les pourcentages étant convertis en valeurs décimales pour un traitement ultérieur.

3. step_2(product_name: str, country_name: str, size: str, materials: list)

La fonction `step_2` traduit les données extraites en identifiants et valeurs utilisables par le modèle prédictif. Elle utilise des fichiers de référence (CSV) pour associer le nom du produit, le pays, et les matériaux à leurs identifiants respectifs. Elle calcule également le poids du produit en fonction de sa taille et de son type en accédant à une table de correspondance des poids. Le résultat est un ensemble de données structurées comprenant l'identifiant du produit, son poids en kilogrammes, le code du pays de fabrication, et une liste d'identifiants pour les

matériaux. Cette étape établit un lien entre les données utilisateur et les formats requis par le modèle.

4. `step_3(product_id: str, mass_kg: float, country_making: str, materials: list)`

La fonction `step_3` génère un jeu de données explicatif pour chaque matériau utilisé dans le produit. Pour chaque matériau, elle crée un ensemble de variables comprenant l'identifiant du produit, son poids, l'identifiant du matériau, et le code du pays de fabrication. Ces informations sont organisées dans un DataFrame Pandas, qui est ajouté à une liste de jeux de données. Cette étape prépare les données nécessaires à la prédiction en structurant les variables explicatives de manière compatible avec les attentes du modèle de machine learning.

5. `step_4(list_Xpred: list, material_percs: list)`

La fonction `step_4` effectue la prédiction de l'impact environnemental du produit en combinant les contributions des différents matériaux. À l'aide d'un modèle pré-entraîné chargé depuis un fichier `.pkl`, elle calcule les prédictions pour chaque matériau en pondérant leurs contributions par leurs pourcentages respectifs. Le résultat final est arrondi pour produire une prédiction entière. Cette étape combine les données explicatives et le modèle prédictif pour fournir un résultat global, offrant ainsi une estimation quantitative de l'impact environnemental du produit.

- `tools.py`

Le module Tools contient les fonctions de base nécessaires pour traiter les données recueillies depuis le programme VBA dans le but d'effectuer la prédiction.

1. `extract_words(sentence: str)`

La fonction `extract_words` a pour objectif de diviser une phrase en une liste de mots individuels. Elle utilise la méthode standard de séparation en Python, basée sur les espaces et autres séparateurs par défaut, pour segmenter une chaîne de caractères en plusieurs unités textuelles. En prenant en entrée une phrase ou un texte sous forme de chaîne de caractères, cette fonction retourne une liste de mots.

2. `get_csv(csv_name: str)`

La fonction `get_csv` permet de charger un fichier CSV en mémoire sous la forme d'un DataFrame, une structure de la bibliothèque Pandas. Ce fichier doit être spécifié par son nom, incluant l'extension, et est recherché dans un chemin relatif défini au sein de l'application. Si le fichier est trouvé et valide, la fonction retourne un DataFrame contenant les données du CSV. La fonction vérifie également que l'argument est une chaîne non vide, et elle soulève des exceptions appropriées (`TypeError`, `ValueError`) en cas de problème.

3. `load_model(model_name: str)`

La fonction `load_model` est conçue pour charger un modèle de prédiction préalablement entraîné et sauvegardé au format `.pkl` (Pickle). En fournissant le nom du fichier modèle, incluant l'extension `.pkl`, cette fonction retourne l'objet représentant le modèle chargé. Des validations sont intégrées pour vérifier que le nom du fichier est une chaîne non vide et qu'il respecte l'extension requise. En cas de problème, des exceptions explicites sont levées (`TypeError` pour un type invalide, `ValueError` pour une extension ou un format incorrect).

4. `get_id_from_name(df: pd.DataFrame, name_column: str, value_to_find: str, id_column: str)`

La fonction `get_id_from_name` effectue une recherche dans un `DataFrame` Pandas pour associer une valeur spécifique d'une colonne donnée à un identifiant stocké dans une autre colonne. En prenant en entrée un `DataFrame`, le nom de la colonne contenant les valeurs à rechercher, la valeur cible, ainsi que le nom de la colonne des identifiants, la fonction renvoie l'identifiant correspondant si une correspondance exacte est trouvée. Si la valeur n'est pas présente, elle retourne `None`.

5. `get_weight_from_product(df: pd.DataFrame, product_type: str, size: str)`

La fonction `get_weight_from_product` permet de calculer le poids d'un produit spécifique à partir d'un `DataFrame` contenant des informations sur les poids des produits par type et par taille. L'utilisateur fournit le type de produit (par exemple, `'chemise'` ou `'jean'`), ainsi que la taille souhaitée (comme `'XS'`, `'S'`, `'M'`, etc.), et la fonction retourne le poids correspondant en kilogrammes. Les poids sont récupérés depuis le `DataFrame` et convertis de grammes en kilogrammes pour une meilleure lisibilité. Si le type de produit ou la taille spécifiés ne sont pas présents dans le `DataFrame`, la fonction retourne `None`.