**7. DATA WRANGLING**

*7.1 Nettoyage du dataset*

**7.1.1 Introduction**

Le volume de données des datasets de nos jours ne cessent de grandir, de même que les erreurs liées à leurs données. Par conséquent, il est normal de devoir effectuer des nettoyages de données de plus en plus minutieux afin de renforcer l’intégrité et la pertinence de ceux-ci en réduisant les incohérences. Eviter les erreurs permettent de rendre les décisions prisent via les données plus précises et mieux avisées.

Le processus de nettoyage des données permet d’identifier et corriger les données altérées, inexactes ou non pertinentes. Ce processus est fondamental afin d’améliorer la cohérence, la fiabilité et la valeur de nos données.

Dans la plupart des cas, l’inexactitude des données viennent de valeurs manquantes dans le dataset, des fautes de frappes ou une donnée se trouvant au mauvais emplacement.

Plusieurs manières de nettoyage sont possible ; Correction des données, renseignement des données ou simplement suppression de celles-ci.

**7.1.2 Analyse**

Nous avons remarqué durant l’analyse de notre dataset certaines incohérences ou erreurs dans celui-ci.

Premièrement dans la colonne APP\_NAME nous avons remarqué que certaines valeurs était incorrect. En effet la colonne comportait énormément de nom d’application composé uniquement de point d’interrogation. Elles ne représentaient pas des noms d’applications.

Dans la colonne Category, certaines valeurs ne représentait pas du tout des catégories et était donc des valeurs incorrect.

Ensuite, la colonne reviews ne comportait pas de virgule et donc était pratiquement uniquement composé de valeurs aberrantes qui était censé être des valeurs avec un maximum de 5.

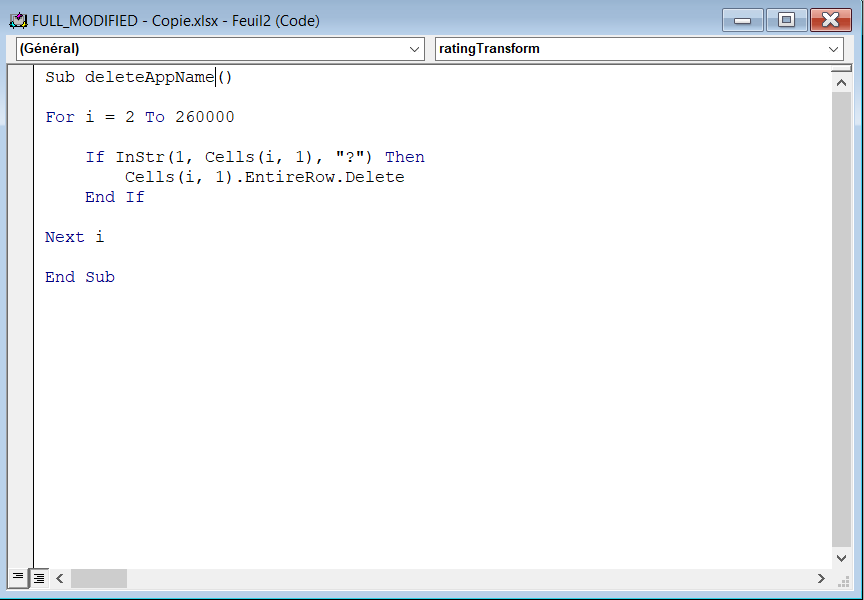
Pour continuer, la colonne Installs n’était pas une colonne en chiffre mais une colonne en « string » composé de chiffre suivi dans la plupart des cas d’un « + ». Un champ « EDUCATION » se trouvait également dans cette colonne.

Enfin, les dernières colonnes ne comportaient aucune donnée et était donc totalement inutile.

**7.1.3 Mise en pratique**

Pour la modification des données du dataset nous avons décidé d’utiliser EXCEL. Effectivement EXCEL fournit un accès à un espace développeur permettant de modifier les données à notre guise via un langage de programmation simple à utiliser appelé VBA.

Pour la première colonne APP\_NAME nous avons décidé de simplement supprimer toutes les lignes dont la valeur de cellule se trouvant dans APP\_NAME était composé d’au moins un point d’interrogation. Pour ce faire nous avons utiliser cette fonction que nous avons créé. Cette fonction permet de parcourir le jeu de données grâce à une boucle allant de la ligne 2 à la ligne 260000 en vérifiant si la valeur de la cellule de la première colonne (« APP\_NAME ») contient ou non un point d’interrogation. Si cela est le cas, la ligne entière est supprimée.



Ensuite la colonne CATEGORY était composé de valeurs incorrectes. Nous avons décidé de supprimer les lignes composés de celles-ci. Les valeurs étaient :

« ) »

« 6 »

« Alfabe ï¿½?ren »

« Islamic Name Boy & Girl+Meaning »

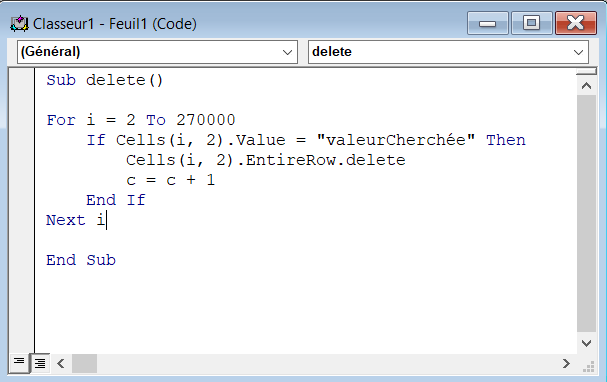
« not notified you followed – »

« Mexpost) »

« Tï¿½rk Alfabesi »

Valeurs vies

Pour ce faire nous avons utilisé une fonction que nous avons créée. Cette fonction permet de parcourir toutes les cellules de la deuxième colonne (« CATEGORY ») du dataset via une boucle. Si la valeur de la cellule correspondait à la valeur que nous voulions supprimer, la ligne entière était supprimée.

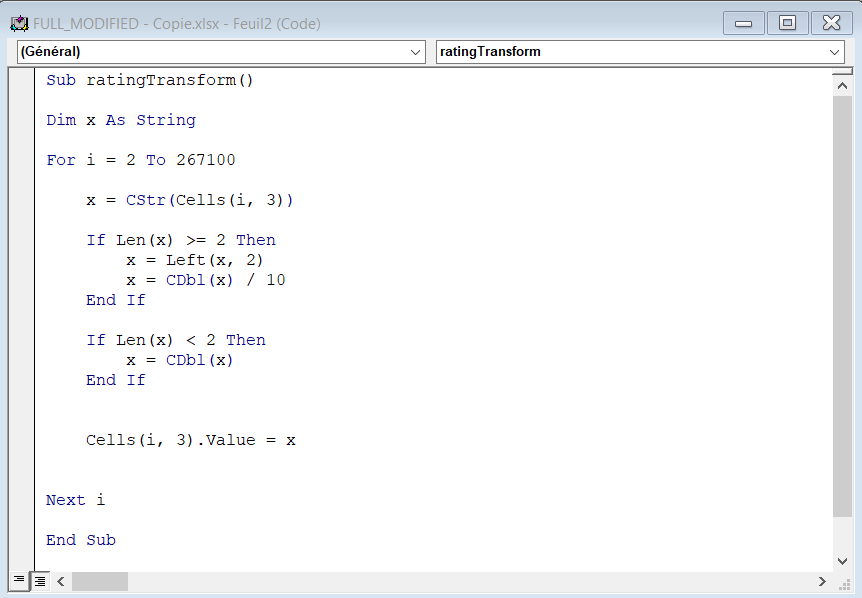


Pour continuer, la colonne reviews devait accueillir la note moyenne sur 5 des applications. Malheureusement il manquait la virgule ce qui nous donnait des nombres très important et pas du tout représentatif d’une note /5. Pour arranger ca nous avons implémenter une fonction. Cette fonction permet de transformer le chiffre en chaine de caractère.

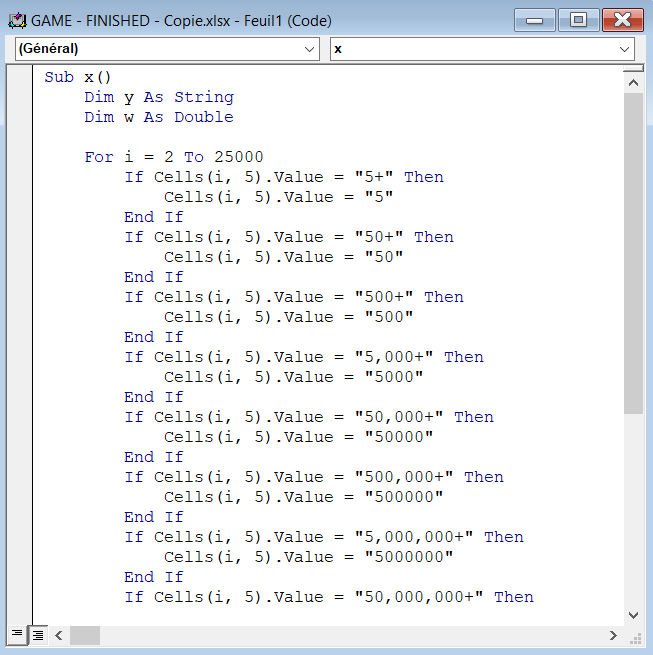
Si la longueur de la chaine est égale ou supérieur à 2, nous retenons les eux premiers caractères de celle-ci, la transformons en double et la divisons en deux.

Si dans le cas contraire, la longueur de la chaine est inférieur à 2, alors nous transformons seulement la chaine en double.

Pour finir nous remettons la nouvelle valeur dans la cellule adéquate.



Concernant la colonne « Installs », les données dans cette colonne étaient des chaines de caractères composé du nombre approximatif de téléchargement accompagné d’un « + ». Notre démarche d’étude se basant principalement sur le nombre de téléchargement, nous étions obligés de modifier cette colonne. Nous avons donc transformé chaque valeur en sa valeur correspondante via une fonction composé de conditionnelle. Nous avons également trouvé une valeur « EDUCATION » dans cette colonne qui était certainement une erreur d’encodage que nous avons simplement supprimé avec la fonction précédente recensé dans la colonne CATEGORY.



*7.2 Mise en forme du dataset*

**7.2.1 Introduction**

Afin de limiter son poids et optimiser son utilisation, un dataset doit être correctement construit. Les modèles de constructions les plus populaires sont des modèles de bases de données, dans lesquelles nous remplaçons les valeurs purement catégoriques via des identifiants, qui seront référencés par la suite dans des tables. Ce système permet d’alléger la table principale en remplaçant tous les champs répétés par des chiffres tandis que la vraie valeur n’est référencée qu’une fois dans sa propre table.

**7.2.2 Analyse**

Notre cas de dataset est simple, nous avons deux colonnes catégoriques, la colonne CATEGORY et la colonne CONTENT\_RATING, que nous pouvons changer en chiffre, pour créer des identifiants qui seront référencés sur des autres tables par après.

Il y aurait alors une table appelée « CONTENT\_RATING » composé d’une colonne identifiante et une colonne composée du label de chaque catégorie de CONTENT\_RATING.

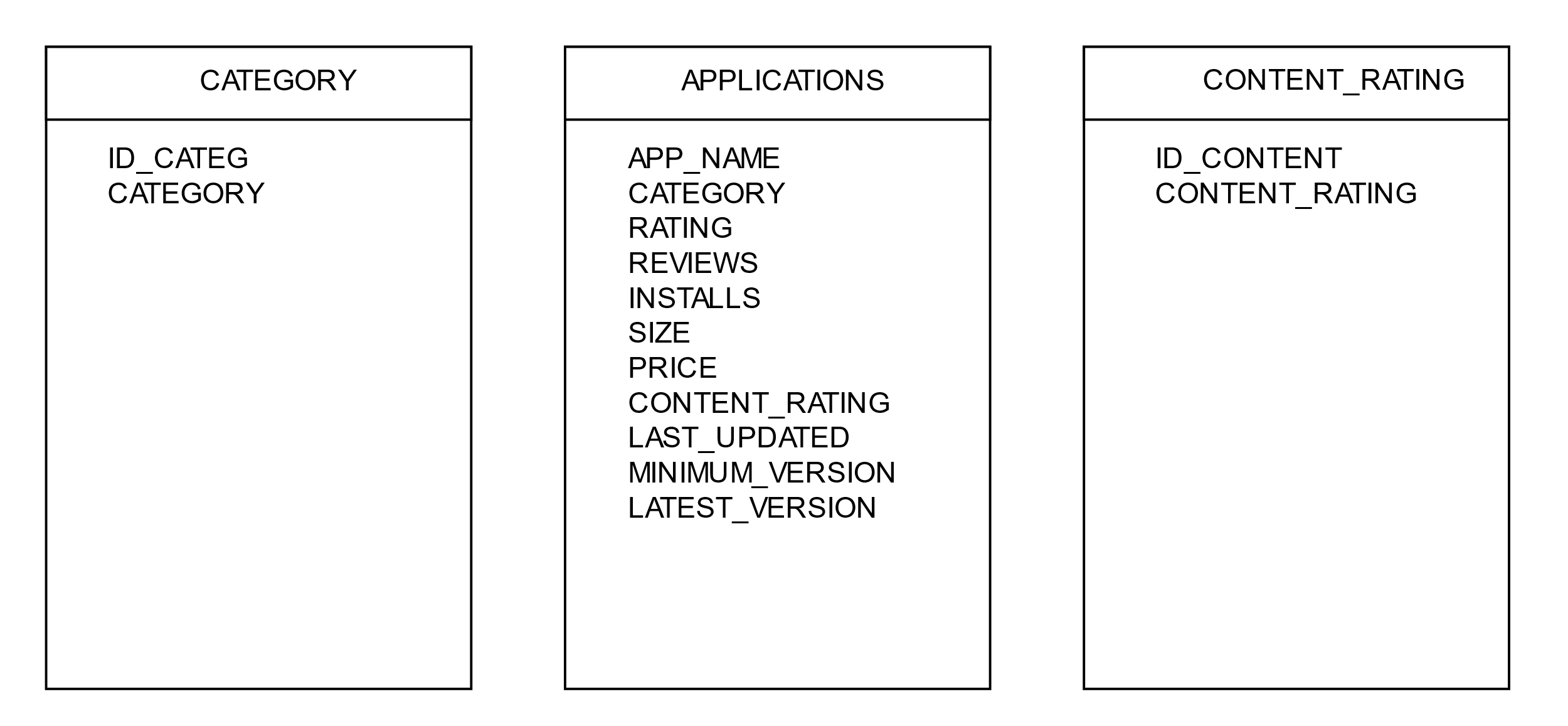
Pour ce qui concerne la table CATEGORY, le système est plus complexe. Possédant dans cette colonne beaucoup de sous-catégorie GAME (telles que « GAME\_ACTION », « GAME\_MUSIC », …), il serait intéressant de créer une étude sur un dataset où toutes les sous catégories de « GAME » seraient rassemblées dans une seule catégorie et une autre étude sur un dataset composé uniquement de toutes les sous catégories « GAME ».

**7.2.3 Mise en pratique**

En ce qui concerne la colonne CONTENT\_RATING, le protocole fut simple, nous avons d’abord créer une table contenant toutes les différentes valeurs de cette colonne et nous les avons accompagnés d’identifiants. Ensuite via une fonction en VBA nous avons remplacé toutes les valeurs de cette colonne par des identifiants correspondant.

Au niveau de la colonne catégorie, comme dit précédemment nous avons décidé de spliter notre dataset en deux. D’un côté un dataset avec les catégorie où les sous catégories « GAME » seraient globalisées dans une seule catégorie et d’un autre côté un dataset où il n’y aurait que les catégories de « GAME ».

La procédure fut la même que pour le CONTENT\_RATING, nous avons créer dans les deux cas une table avec les noms des catégories et un identifiants leur correspondant. Dans la table principale nous avons substituer les valeurs initiales par les identifiants correspondants.



Nous pourrions considérer cet ensemble de tables comme une base de données. Dans ce cas, dans la table CATEGORY nous pouvons considérer la colonne ID\_CATEG comme clef primaire. La colonne CATEGORY comporte les labels des catégories.

La table CONTENT\_RATING est similaire à la table CATEGORY. Nous pouvons considérer la colonne ID\_CONTENT comme clef primaire. La colonne CONTENT\_RATING comporte les labels.

**8. VISUALISATION**

*8.1 Introduction*

La visualisation des données représente les méthodes utilisées afin de présenter les données sous formes de schémas visuels. Ils peuvent être présenté sous formes de différents graphiques, peuvent être composés de différentes couleurs, etc… Le tout utilisé le mieux possible pour en faire ressortir l’essentiel.

La partie de visualisation est très importante car ce sera la partie la plus visuelle du projet. Grâce aux schémas nous pourrons présenter les points les plus importants et plus marquants à des personnes qui auront connaissance ou non de notre projet.

*8.2 Choix du logiciel de visualisation*

**8.2.1 Introduction**

De nos jours, les moyens informatiques qui permettent de visualiser des données sont de plus en plus nombreux. Les solutions nous permettant de présenter notre dataset en deux ou trois dimensions sont en majeure partie toutes très performantes. Entre les outils microsoft, tableau et qlik les possibilités étaient grandes. Ainsi pour choisir le meilleur logiciel de visualisation de données, nous avons décidé de procéder à une analyse suivie par une élimination.

**8.2.2 Analyse**

Le logiciel tableau était un choix pertinent car beaucoup utilisé dans le monde de l’entreprise. Il présente des performances et des ressources très intéressantes. De plus il possède une version gratuite appelée « Tableau Public ». Le seul « inconvénient » de ce logiciel était que la version « public » obligeait à mettre nos données en ligne, ce qui ne posait pas de problème puisque notre dataset provenait d’un dataset trouvé sur le net. L’utilisation de ce logiciel était intuitive mais mettait un peu de temps pour comprendre son utilisation et malheureusement seul un membre de notre groupe connaissait son utilisation.

Le deuxième choix potentiel était le logiciel POWER BI proposé par la société microsoft. Appris et utilisé à un cours durant notre bachelier de spécialisation, nous avions tous les compétences et connaissances pour mener à bien notre partie de visualisation grâce à ce logiciel. L’inconvénient du prix n’entrait pas en rigueur dans notre situation car nous avions accès à une version étudiante gratuite du logiciel.

Notre troisième choix était l’application web QLIK. Également apprise durant notre bachelier de spécialisation, nous avions tous les trois, tout comme le logiciel de la société microsoft, les compétences et connaissances requises pour mener à bien notre projet. Le prix n’entrait pas non plus en rigueur dans notre situation car nous avons reçu un accès étudiant à celui-ci.

**8.2.3 Conclusion**

Après débat au sein de notre groupe les deux logiciels les plus adapté pour effectuer notre projet était bien sur POWER BI et QLIK. Les deux logiciels présentaient des performances similaires et ne se démarquaient pas réellement l’un de l’autre, de plus nous voulions travailler sur un seul logiciel pour limiter la migration des données qui aurait pu être handicapante. Nous avons donc décidé de voter pour en éliminer un des deux. Après le vote le choix s’est porté sur QLIK à l’unanimité.

*8.3 Prérequis*

**8.3.1 Introduction**

Dans cette partie nous allons tout d’abord présenter les colonnes via des fonctions de leur moyenne, somme, etc… Et y créer des mesures. Nous pourrons réutiliser ces mesures afin de démontrer par après, les différences au niveau des catégories et du contenu. Les chiffres présentés dans cette section seront des données calculées sur base du dataset GLOBAL. Les fonctions seront bien sûr créées au même titre sur base des données du dataset GAME.

**8.3.2 Les mesures**

Pour avoir une utilisation plus pratique de notre visualisation nous avons utilisé des « mesures ». Une mesure dans Qlik est une valeur calculée sur base d’une fonction. Par exemple celle-ci peut, en ayant spécifié la colonne, donné la valeur la plus grande. Elle peut également servir de calcule. Ces mesures peuvent être attribuées à des dimensions afin de visualiser une certaine mesure dans une certaine dimension.

Exemple : Nous voulons le nombre de téléchargements par catégories. La mesure à créer est le nombre de téléchargements et la dimension est la catégorie. Nous créons donc la mesure grâce à une fonction « sum » et l’associons à la dimension « CATEGORY » sous forme de graphique en bar.

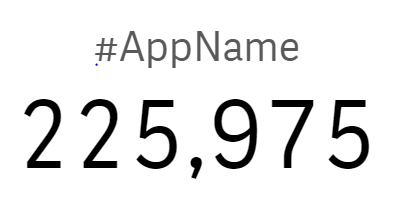
Dans notre cas les mesures que nous allons créer sont basées sur les fonctions suivantes :

* Sum([Colonne]) : Permet de faire la somme de toute la colonne spécifiée.
* Max([Colonne]) : Permet de faire ressortir le plus grand nombre de la colonne spécifiée.
* Min([Colonne]) : Permet de faire ressortir le plus petit nombre de la colonne spécifiée.
* AVG([Colonne]) : Permet de faire ressortir la moyenne de tout les nombre de la colonne spécifiée
* Count([Colonne]) : Permet de faire ressortir le nombre de lignes dans la colonne.

Grâce à ces fonctions nous allons pouvoir faire ressortir beaucoup de données intéressantes sur lesquelles nous pourrons jouer via des dimensions telles que CATEGORY et CONTENT\_RATING.

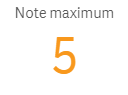
**8.3.3 App Name**

Nous avons décidé de seulement calculer le nombre de lignes de la colonne APP NAME permettant ainsi de pouvoir attribuer, associé à une dimension, le nombre d’applications qui y correspond.

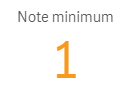
 ****

**8.3.4 Rating**

Pour la colonne Rating, regroupant les notes sur 5 des applications nous avons décidé de mettre en avant le minimum, le maximum et la moyenne des notes.

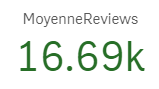
 

**8.3.5 Reviews**

Pour la colonne Reviews, composée du nombre de notes attribuées à l’application, nous avons décidé de créer, comme pour la colonne précédente les mesures maximums, moyenne et minimum.







Nous avons également choisi de créer une somme des Reviews afin de pouvoir l’attribuer à une dimension.

****

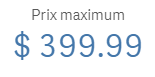
**8.3.6 Installs**

La colonne « Installs » regroupant le nombre d’installations par application est une colonne essentielle à notre TFE. Effectivement nous allons baser nos graphiques principaux sur le nombre de téléchargements sous certaines dimensions.



**8.3.7 Prix**

La dernière colonne sur laquelle nous allons créer des mesures est la colonne prix, contenant le prix de chaque application. Nous allons créer un maximum, un minimum et une moyenne.

*8.4 Dashboard*

**8.4.1 Introduction**

Un dashboard est un outil d’évaluation qui va nous permettre, via des indicateurs, de faire ressortir les données importantes à présenter. Les dashboards sont utilisés afin de présenter les données importantes à faire ressortir. Les indicateurs, appelés aussi KPI, peuvent être présenté sous différentes formes comme ; graphique en bar, jauge, …

Ces dashboards peuvent être opérationnelles ou décisionnelles. Les dashboards opérationnelles seront attribuées aux départements avec des données centré sur le département en lui-même tandis que les dashboards décisionnelles eux seront plutôt destinés aux managers, CEO, … regroupant les données permettant de résumer la situation actuelle.

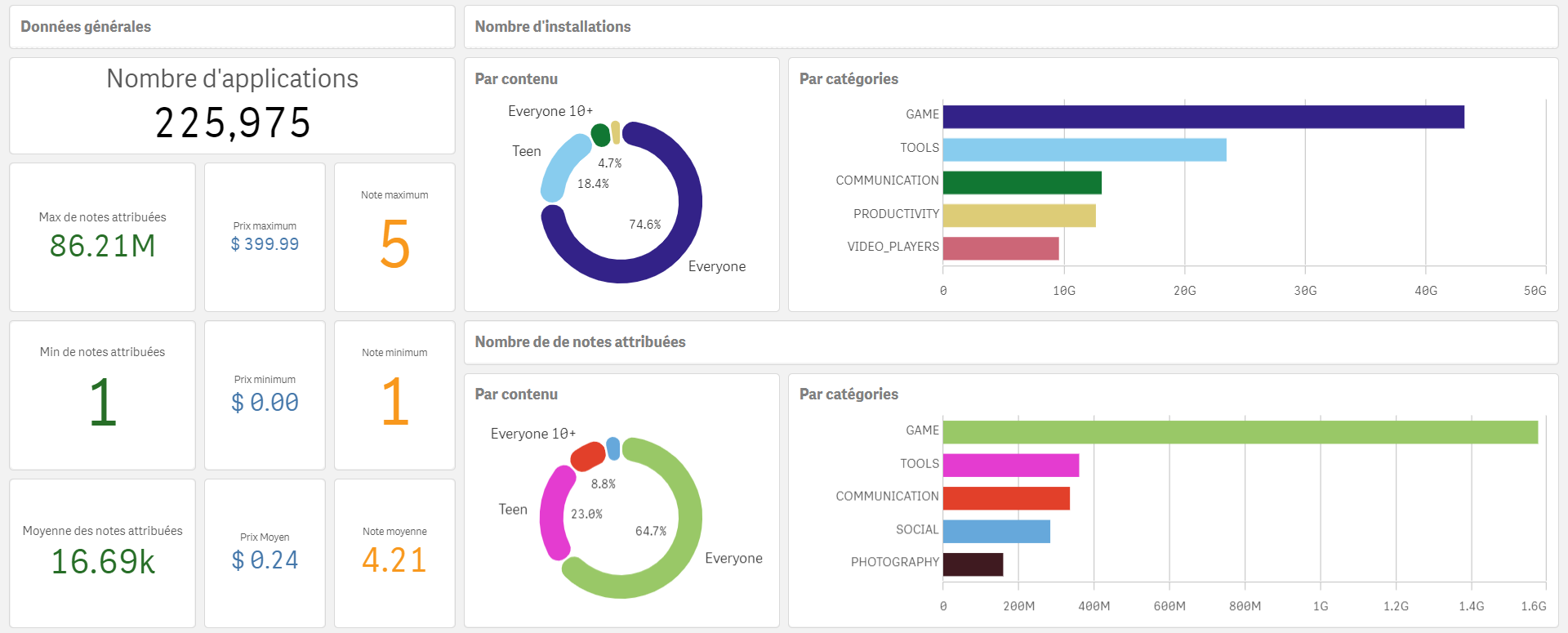
Dans notre cas, ayant peu de données diverses, notre choix vers un dashboard décisionnelle était le plus adapté. Nous allons tout de même, via plusieurs KPI accompagnés de mesures préalablement créées, mettre en avant quatre dashboards.

Les deux premiers dashbaords seront basés sur le dataset  « GLOBAL », expliqué dans la partie de nettoyage des données. L’un présentera les données générales du dataset tandis que l’autre permettra de modeler les visualisations de notre dataset via des filtres.

Ensuite, les deux suivants seront basés sur le dataset « GAME ». Il présentera l’un et l’autre les mêmes spécificités que les deux premiers mais basés sur le dataset « GAME ».

**8.4.2 GLOBAL – Vision d’ensemble**

Afin d’avoir une vue d’ensemble sur notre dataset nous avons implémenté un dashboard permettant cette vue. En effet, nous avons regroupé les mesures créées dans le point précédent qui nous permettait d’obtenir dans la majorité des cas ; le minimum, le maximum et la moyenne. Ces points vont nous permettre d’une part, pouvoir repérer les caractéristiques moyenne des applications et d’une autre part pouvoir déterminer dans un premier temps les données qui ne seront pas assez utile pour être utilisé dans notre « dashboard filtrer », qui sera expliqué au point suivant.



Voir annexe

Nous pouvons constater qu’un groupement de données générales se trouve sur la gauche de notre dashboard. Ces données, créées via nos mesures, permettent de visualiser dans l’ensemble la moyenne, le maximum et le minimum de toutes nos données.

Ensuite dans la partie de droite supérieur nous avons, un graphique en camembert contenant les contenus les plus téléchargés à gauche et un graphique en barres des catégories les plus téléchargées à droite.

Dans la partie droite inférieur, nous pouvons visualiser un graphique en camembert de nouveau, contenant les contenus ayant le plus de notes attribuées à gauche et un graphique en barre des catégories ayant obtenu le plus de notes à droite.

*Que pouvons-nous constater ?*

Premièrement dans les données à gauche, le nombre de notes attribuées se situe entre 1 et 86.21M. La moyenne de celle-ci se situe à 16.690 notes. Au niveau du prix, celui-ci varie entre 0.00$ (qui sont les applications gratuites) et 399.99$ (qui est l’application la plus chère du dataset). La moyenne quant à elle est de 0.24$. Cette moyenne très basse peut être expliquée par le nombre très grand d’applications gratuites dans le dataset. Enfin au niveau des notes, sans grande surprise les notes fluctuent entre 1 et 5. La note moyenne de 4,21 est malgré tout très haut.

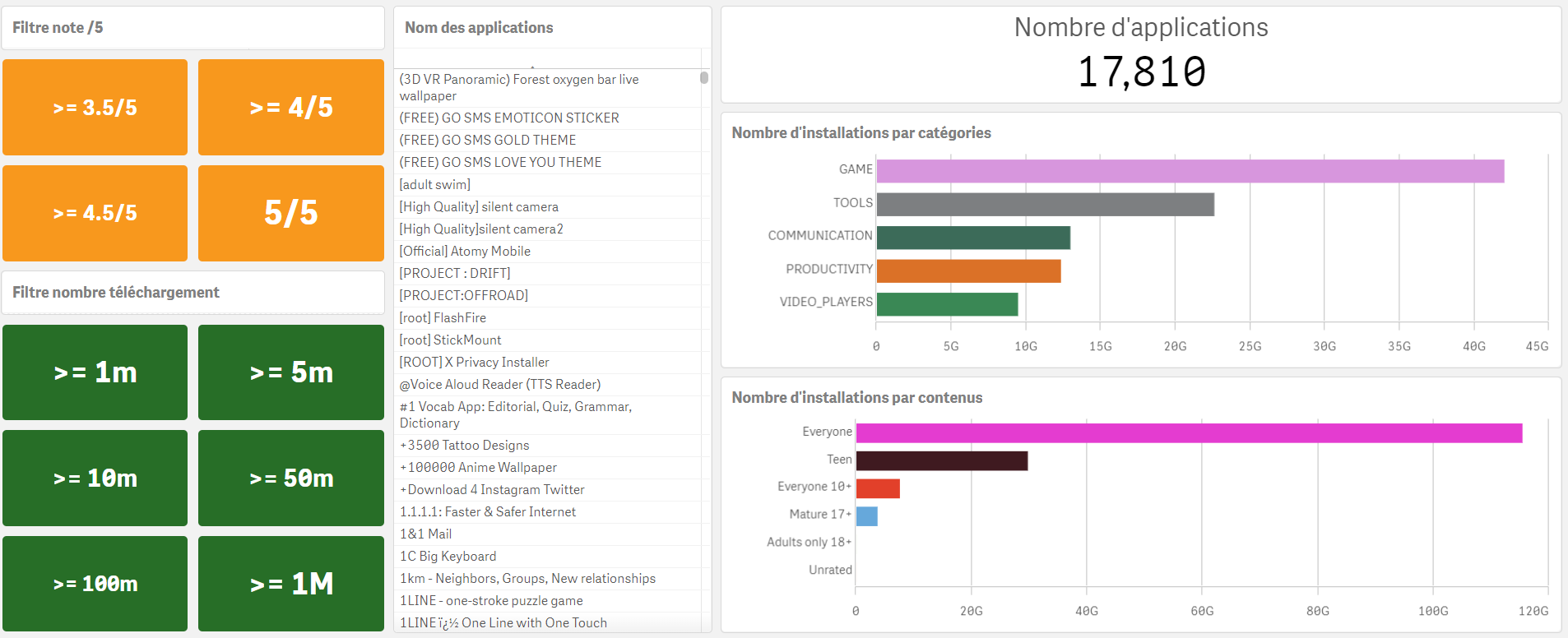
Ensuite dans nos données de droite, nous pouvons constater qu’autant au niveau du nombre de notes attribuées qu’au niveau du nombre d’installations, les données ressorties sont presque les mêmes. Au niveau des catégories, dans notre top 3 ; « GAME », « Tools » et « communication », nous pouvons constater une démarcation de la part de la catégorie « GAME ». Au niveau du contenu nous constatons pareil. Le contenu le plus présent au niveau des installations et des notes attribuées est « Everyone », suivi par « Teen » et « EVERYONE +10 ».

*Et maintenant ?*

Une des ressources de qlik que nous apprécions particulièrement est sa réactivité. En effet, lorsque nous cliquons sur une donnée d’un graphique, tout le dashboard va adapter ses données en fonction de la sélection. Un filtre est en réalité appliqué. Ceci va nous permettre de visualiser les données plus spécifiques telles que toutes nos données générales uniquement avec la catégorie GAME ou avec le contenu « EVERYONE +10 ».

**8.4.3 GLOBAL – Filtre**

Maintenant que nous avons nos données générales et une meilleure visualisation de ce qui se trouve sur le marché de l’android store. Nous pouvons réaliser un dashboard qui va nous permettre de cibler, via plusieurs critères, quelles applications fonctionnent le mieux. Nos critères sont simples ; le nombre de téléchargements ainsi que la note. Via des boutons créés dans QLIK qui, en les sélectionnant, appliquent des filtres nous allons pouvoir filtrer nos données. Une fois nos données filtrées nous pourrons voir, les catégories et les contenus dans lesquelles les applications ont le mieux fonctionné.



Dans ce dashboard « à filtre » nous avons donc dans notre partie de gauche les filtres (qui sont en réalités des boutons). Une fois sélectionnés, ils appliqueront une condition dans le jeu de données n’affichant uniquement les données respectant les conditions. Par exemple si je clique sur le bouton « >= 3.5/5 » un filtre s’appliquera et n’acceptera que les données où la note est de au moins 3.5/5.

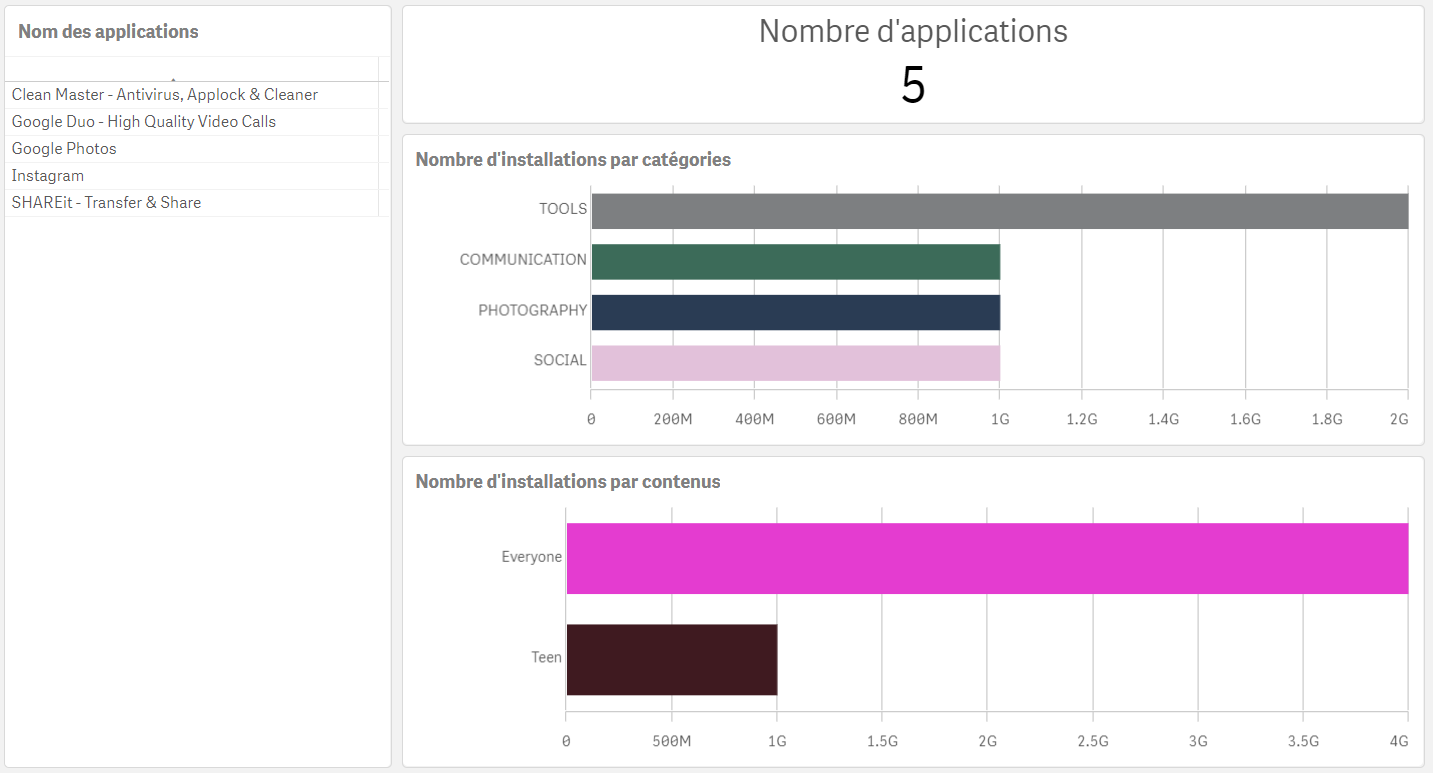
Les filtres sont :

* Au niveau supérieur 4 possibilités pour filtrer sur la note.
* Au niveau inférieur 6 possibilités pour filtrer sur le nombre minimum de téléchargement.

Au milieu du dashboard nous pouvons retrouver le nom des applications. Quand le jeu de données n’est pas filtré il est peu intéressant de visualiser ce tableau, en effet. Mais lorsqu’il est correctement filtré nous pouvons en faire ressortir les applications ayant eu le plus de succès au niveau des notes et de nombre de téléchargements. Ainsi nous pourrons également en faire ressortir, via la partie droite du dashboard, leurs catégories et leurs contenus.

*Démonstration*

Nous souhaitons avoir les applications ayant été téléchargées plus d’un milliard de fois et ayant une note d’au moins 4.5/5.

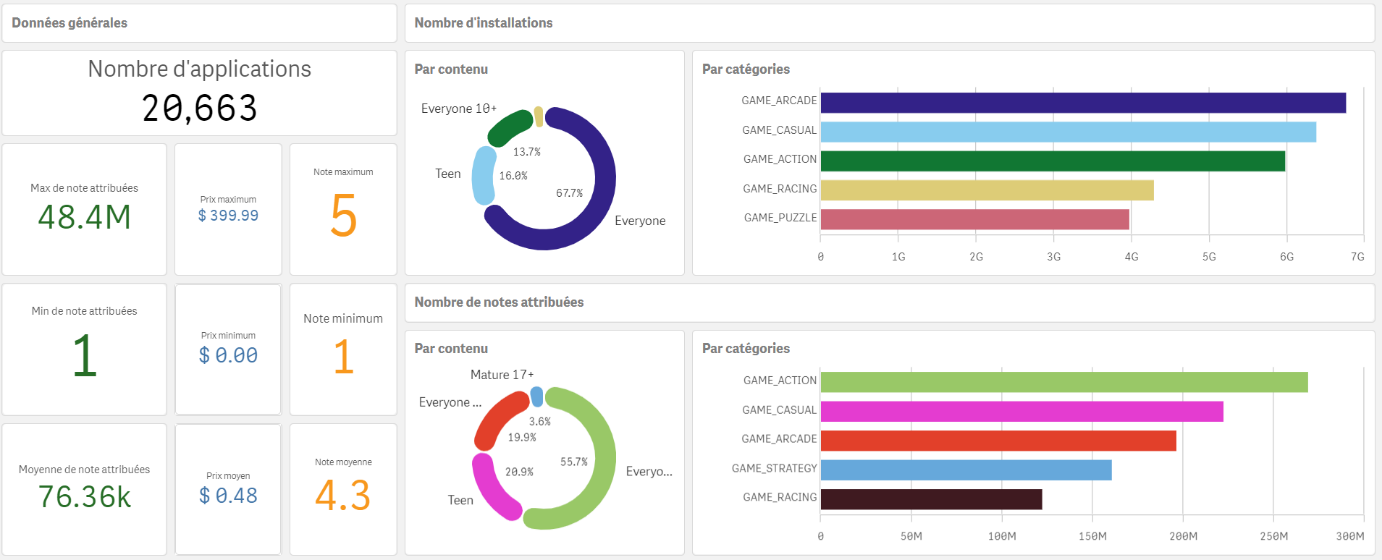


Une fois filtré nous pouvons voir que notre dashboard a bien changé. Il reste donc un total de 5 applications respectant ces données.

**8.4.4 GAME – Vision d’ensemble**

Comme précédemment dit à plusieurs reprises, nous avons axé un deuxième dataset crée à partir du premier. Dans ce dataset seules les sous catégories de la catégorie « GAME » s’y retrouvent. Les jeux vidéo étant très présent à l’heure d’aujourd’hui, nous avons trouvé intéressant de, via les données qui nous étaient proposées, faire ressortir les sous-catégories « GAME » et les données des applications qui fonctionne le plus dans l’android store.

Nous sommes partis du même « template » que pour le dataset axé « GLOBAL » afin d’obtenir une meilleure vue d’ensemble sur notre dataset.



Nous pouvons constater que la présentation est similaire. A gauche les données générales et à droite des graphiques sur le nombre d’installations et le nombre de notes attribuées.

*Que pouvons-nous constater ?*

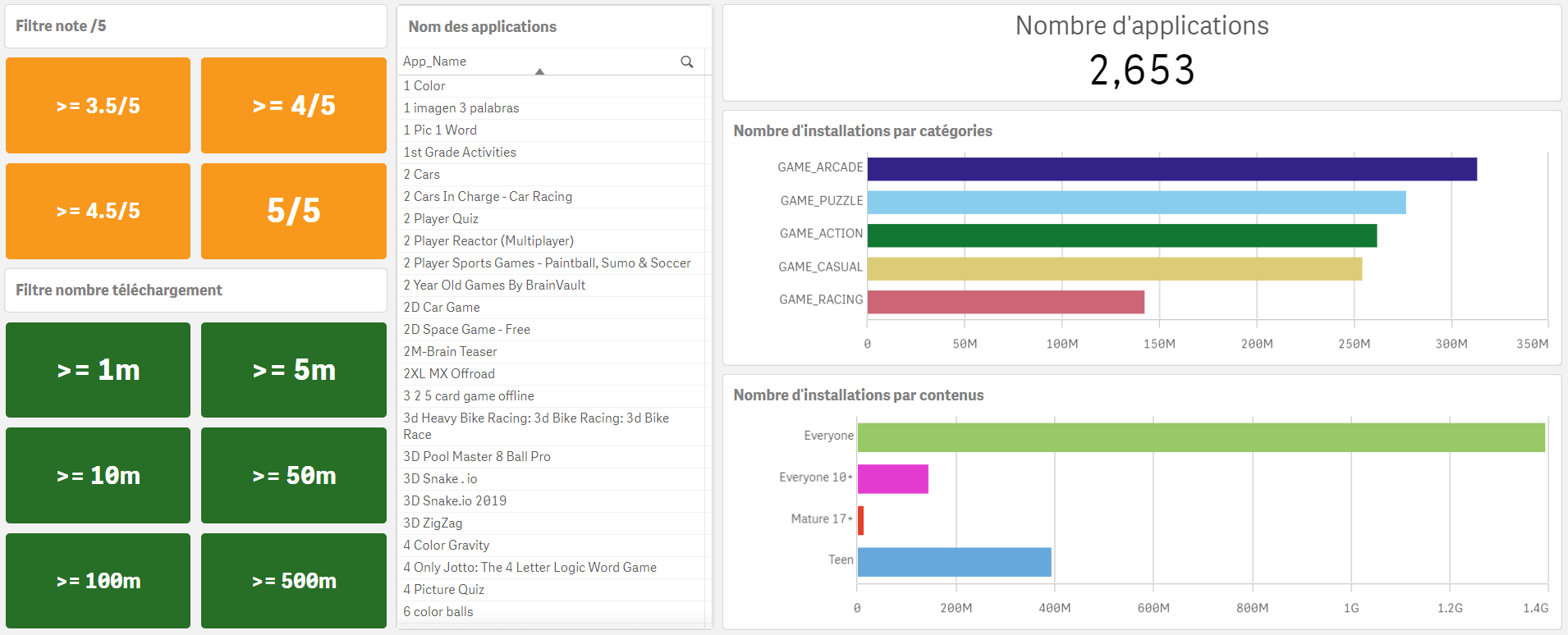
Dans ce dashboard, le nombre d’installations au niveau du contenu « EVERYONE » est toujours aussi écrasant. Suivi de loin par « TEEN » et « EVERYONE +10 ». Au niveau de la catégorie, Le top 3 est « serré » et ne se démarque que par 0.7G installations de différence entre le premier et le 3ème.

Ensuite, nous pouvons constater que, par nombre de notes attribuées, au niveau du contenu la démarcation « Everyone » est moins écrasante. Il y a plus de notes attribuées pour les contenus « Teen » et « Everyone +10 ». Egalement au niveau des catégories, nous avons plus de notes attribuées au niveau de la sous catégorie « ACTION ».

Malgré cela, le top 5 des catégories reste similaire autant au niveau du nombre de notes attribuées qu’au niveau du nombre d’installations.

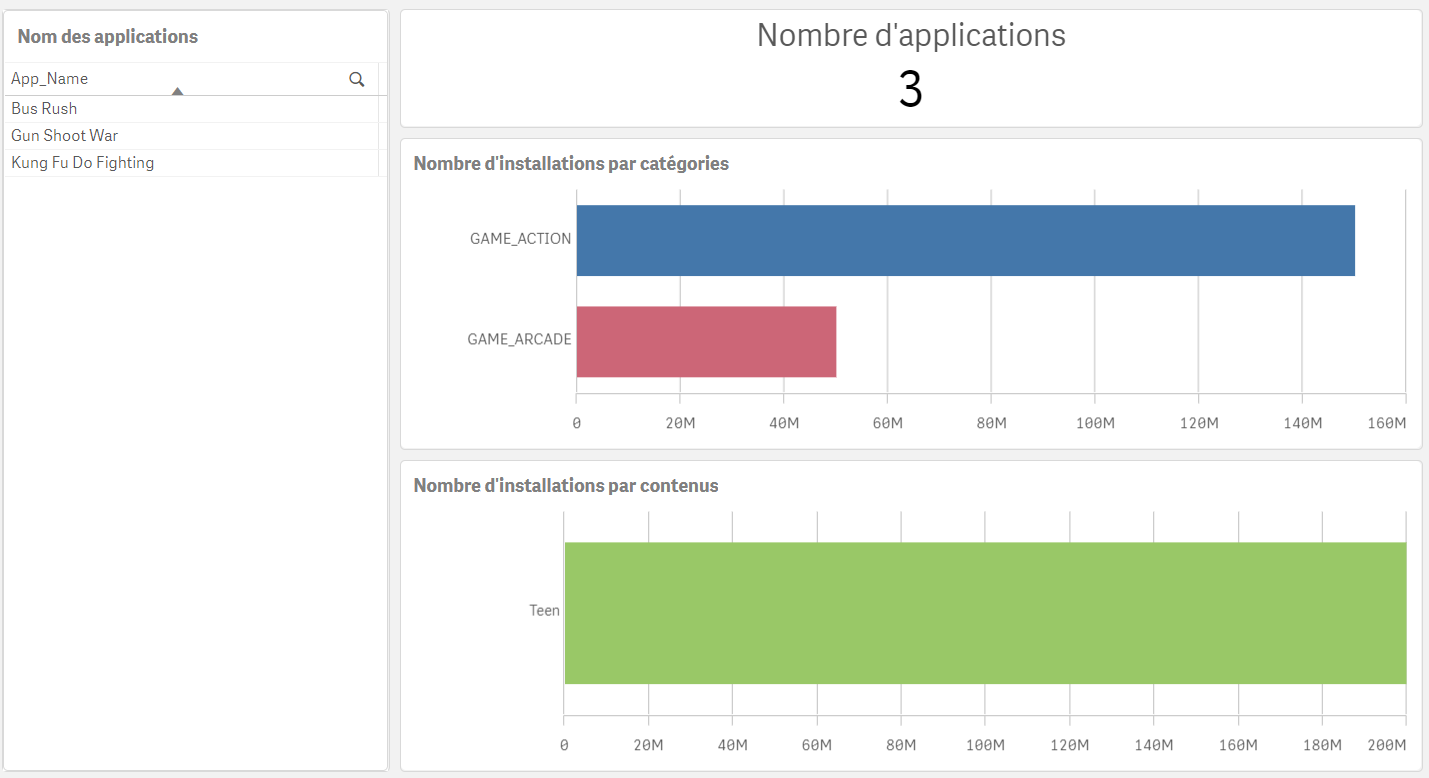
**8.4.5 GAME – Filtre**

A nouveau, nous avons décidé de garder le même « template » pour une utilisation plus pratique de notre visualisation « filtre ». Nos filtres permettent toujours de cibler les caractéristiques que nous souhaitons faire ressortir et mettre en avant de notre dataset.



Comme pour le dashboard « GLOBAL – Filtre », nous avons une partie de gauche contenant les filtres pouvant être appliqué, le nom des applications au centre et enfin notre nombre d’applications, le nombre d’installations par catégories et le nombre d’installations par contenu à gauche.

*Démonstration*



Dans le graphique ci-dessus, nous avons mis en avant trois filtres. Nous avons cherché les applications du contenu « TEEN » ayant au moins 4/5 de note et plus de 50 millions de téléchargements. En réponse à notre recherche nous pouvons constater que les réponses dans notre datasheet sont minces. Seulement 3 applications et deux catégories ressortent.

**8.4.6 Conclusion – GLOBAL**

Après une analyse et un jeu de visualisation sur nos deux dashboards (Général et Filtre) nous pouvons déduire plusieurs choses.

Dans un premier temps, sur un plan global sans appliquer de conditions les catégories qui ressortent autant au niveau du nombre de téléchargements qu’au niveau du nombre de notes attribuées sont « GAME », « Tools » et « Communication ». Notre top 3 pourtant formé nous pouvons remarquer que la catégorie « GAME » écrase complètement les autres. Nous pourrions donc déduire que les applications récoltant le plus de notes et d’installations se retrouvent dans la catégorie « GAME ».

Ensuite au niveau du contenu sans grande surprise le contenu « EVERYONE » est le plus populaire. Cela pourrait être provoqué par une absence de précision lors de l’upload de l’application. En effet, il pourrait laisser paraitre que peu de développeurs qui postent leur application spécifie correctement le contenu de celle-ci. Nous avons notamment trouvé des jeux de guerre, zombies, etc.. Dans le contenu pour tous. Hors ce contenu devrait être dans la catégorie +16 comme dans les normes des autres plateformes de jeux vidéo.

Maintenant appliquons nos filtres, car effectivement une application beaucoup téléchargée et beaucoup notée est une application faisant beaucoup de vues, certes, mais il faut également que les notes soient positives pour pouvoir parler d’une application « à succès ».

Une fois filtré, nous pouvons remarquer les géants se retrouvant dans le haut du classement avec des installations dépassant le milliard. Par exemple Instagram et Google Photos. Si nous nous contentons « seulement » d’au moins 100 millions de téléchargements et d’une note d’au moins 4.5/5 nous pouvons remarquer qu’il ne reste plus que 80 applications. La catégorie étant loin devant cette fois est « TOOLS ».

Unanimement, au niveau du prix, il y a, hors application spécifique et en majeure partie, des applications gratuites. Ce qui pourrait nuire à la popularité d’une application s’il celle-ci se retrouvait payante, face à toutes ces concurrentes gratuites.

Nous pouvons conclure que dans la moyenne la catégorie regroupant le plus d’installations est la catégorie « GAME » lorsque aucun filtre n’est appliqué.

**8.4.7 Conclusion – GAME**

Au niveau des sous catégories de jeux vidéo présentes dans le dataset, notre top 3 des sous catégories, composé de « ARCADE », « CASUAL » et « ACTION », est le même autant au niveau du nombre de notes attribuées que du nombre d’installations.

Le contenu « EVERYONE » est écrasant, autant au niveau du nombre de téléchargements que du nombre de notes attribuées. Comme dit précédemment on peut apercevoir une petite altération au niveau du nombre de notes attribuées. Il pourrait sembler que plus de monde ait noté les applications du contenu « TEEN » et « EVERYONE +10 » ou qu’il y ait moins de notes attribuées dans le contenu « EVERYONE ».

Malgré l’application des filtres, nous pouvons constater que les sous catégories restant en haut du classement sont « ARCADE », « ACTION » et « CASUAL ». Le top 5 ayant peu de mouvement lors d’application de filtres de notes, nous allons plutôt nous baser sur le nombre de téléchargements.

Dans les applications les plus téléchargées, nous avons une grande démarcation de la sous-catégorie « CASUAL » lorsque que nous demandons les applications ayant plus de 500 millions de téléchargements.

**9. MODELISATION STATISTIQUE**

*9.1 Introduction*

La modélisation statistique est une manière mathématique d’approximer la réalité. Autrement dit, elle permet d’expliquer les processus qui créent les données. En plus de cela, elle permet également de réaliser des prédictions à partir de ces approximations.

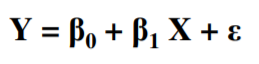
Dans un modèle statistique, il y a, dans la majeure partie des cas, « une variable dépendante » et une ou plusieurs « variables indépendantes ». Aussi appelées « variable à expliquer » et « variable explicative », qui ont plus de sens.

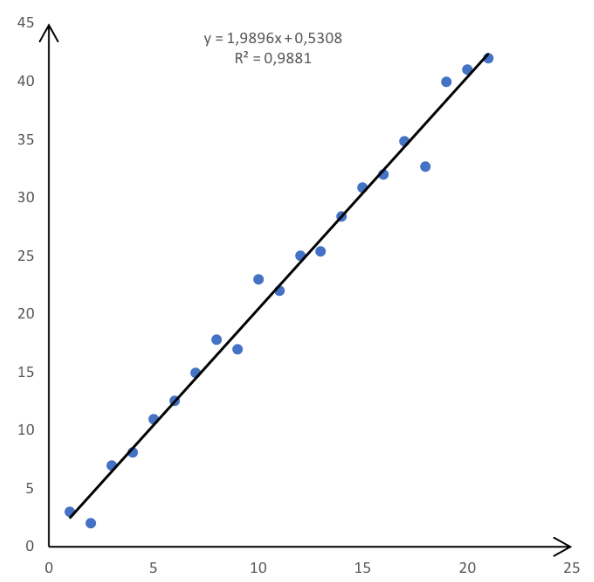
En effet, grâces à nos modèles nous allons pouvoir expliquer quelles « variables indépendantes » prédisent la variable « dépendante ».

Plusieurs modèles existent : Régression linéaire simple, régression linéaire multiple, régression logistique, …

*9.2 Les modèles*

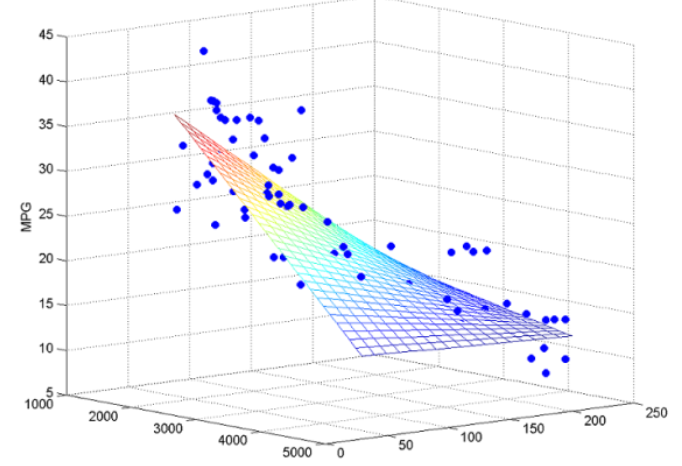
**La régression linéaire simple** est un modèle statistique utilisé lorsque que nous avons une seule variable indépendante et nous voulons savoir si elle prédit ou non notre variable dépendante.



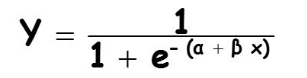
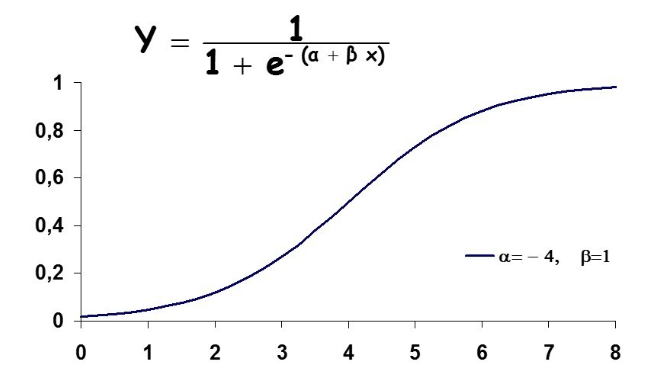
****

**La régression** **linéaire multiple** est un modèle statistique utilisé lorsque nous avons plusieurs variables indépendantes et nous voulons savoir si elles prédisent ou non notre variable dépendante.





**La régression logistique** est un modèle statistique dans lequel notre variable dépendante sera une variable booléenne. C’est-à-dire qu’elle ne peut prendre comme valeur que 0 ou 1 (Non ou oui). Ce modèle va permettre, via la variable indépendante, de prédire la probabilité qu’un élément va se produire ou non.



*9.3 Choix du logiciel*

Les choix se présentant à nous était large, nous avons néanmoins voulu intégrer des connaissances acquises par nous même au sein du projet plutôt que d’utiliser R, déjà utilisé dans d’autres parties. Nous avons donc choisi de nous orienter vers GRETL.

GRETL est un logiciel, « open source » et programmé en C, de statistiques. Sa première version est sortie en 2000. Il peut être utilisé autant en ligne de commandes via le langage « hansl » qu’en interfaces graphiques. Le logiciel paru plusieurs fois dans les journaux de sciences tel que dans le ‘’Journal of Statistical Software’’ en 2008, vendant ses mérites. Sa communauté est forte, il possède même une conférence se produisant tous les 2 ans depuis 2009.

La raison de ce choix est que GRETL est très intuitif, facile à utiliser et performant.

*9.4 Analyse*

Nos datasets, GAME et GLOBAL, sont composé de 11 colonnes chacun. Dans ces colonnes, aucune ne comporte de valeur booléenne. Donc nous pouvons déjà en déduire que le modèle à régression logistique est à éliminer. Il nous reste donc nos deux modèles de régression linéaire. De toute évidence, vu le nombre de colonnes que nous avons, un système de régression linéaire simple n’est pas envisageable.

Nous voilà donc avec notre système de régression linéaire multiple. Nous allons devoir prendre en compte que nous avons des valeurs catégoriques pour « CATEGORY » et « CONTENT\_RATING ». Ces colonnes devront être modifiées dans GRETL pour stipuler qu’il s’agit de variables discrètes.

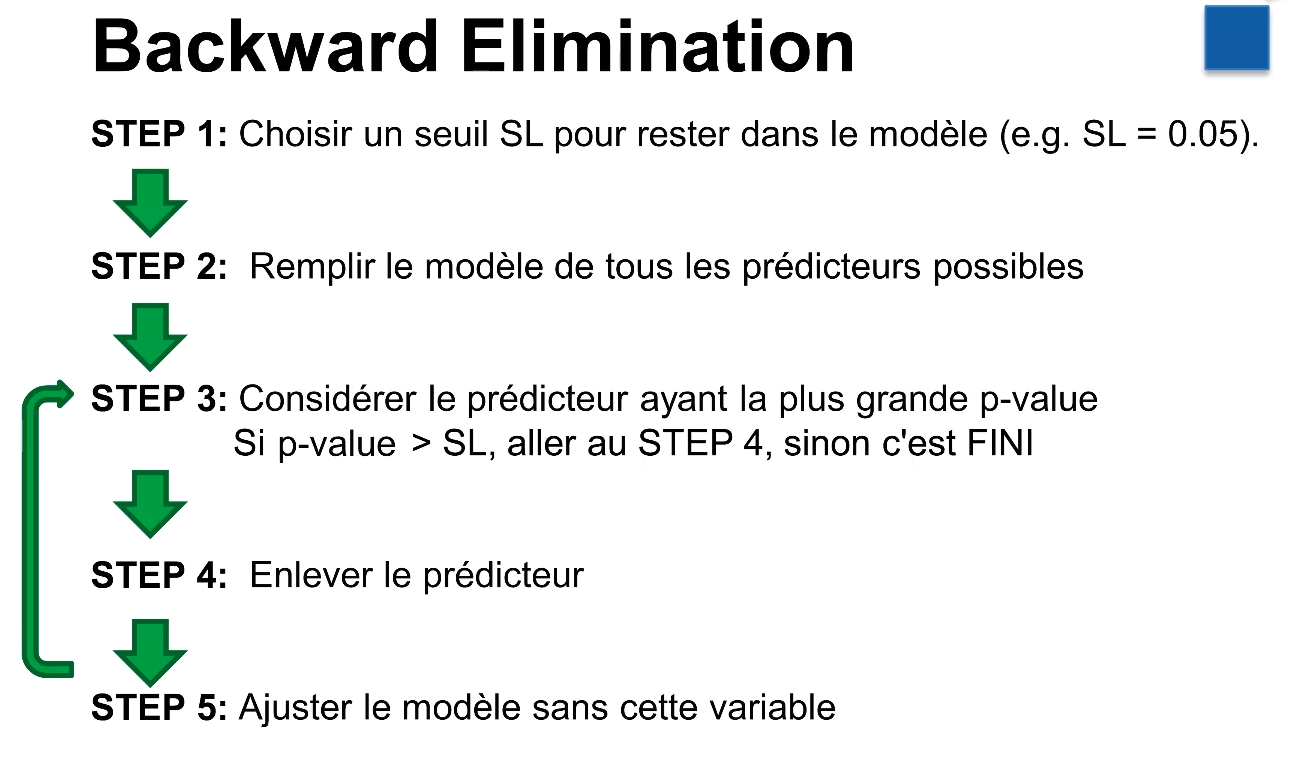
Nous allons donc prendre comme variable dépendante le nombre d’installations et comme variables indépendantes, toutes les autres. Nos variables discrètes devront être créées dans GRETL via la fonction « variables discrètes ».

*9.5* *Choix de la méthode de construction du modèle*

Il existe plusieurs méthodes de construction de modèles ; All-in, Backward-elimination, Bidirectional Elimination, …

Chacune ont leurs points forts et leurs points faibles, malheureusement nous ne connaissons qu’une méthode de construction qui est la « BACKWARD-ELIMINTION ». Nous allons donc utiliser cette méthode.

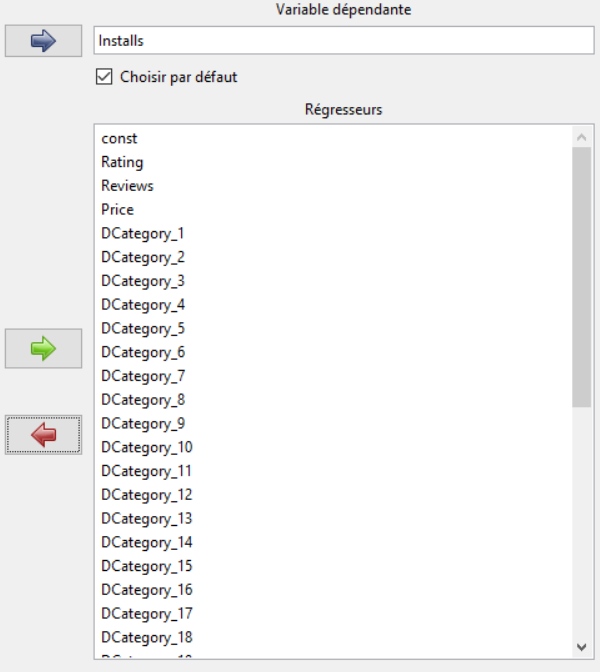
Comment fonctionne-t-elle ?



*9.6 Mise en pratique – Dataset GLOBAL*

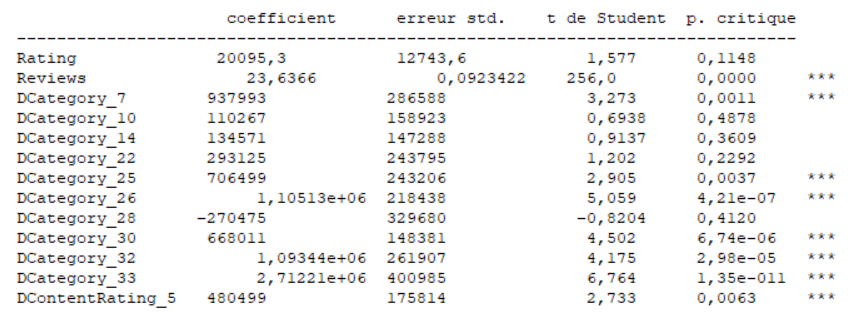
Après avoir choisi comme seuil 5% et spécifier le modèle dans GRETL, il nous suffit de déclarer la variable dépendante (Installs) et les variables indépendantes. En ce qui concerne les « variables discrètes », il faut respecter la règle de ne prendre que les n-1. Dans notre cas nous avons 34 variables discrètes de CATEGORY, nous n’en prenons que 33. Dans content nous avons 6 variables discrètes, nous n’en prenons que 5.

Les paramètres dans GRETL devraient ressembler à ceci :

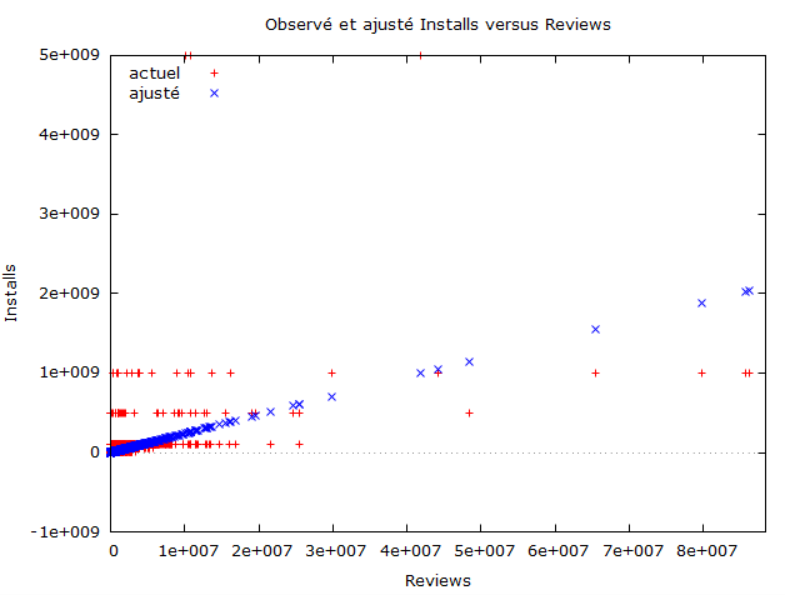


Une fois les répresseurs, nous pouvons valider notre modèle. Nous obtenons, via GRETL, une fenêtre indiquant les p-values des variables indépendantes. Maintenant il ne nous reste plus qu’à appliquer le modèle « BACKWARD-ELIMINATION ». Il nous faut donc éliminer la plus grande p-value et refaire le modèle jusqu’à ce que toutes les p-value respecte le seuil de 5%.

Une fois fini, la méthode du BACKWARD ELIMINATION nous donne ceci :

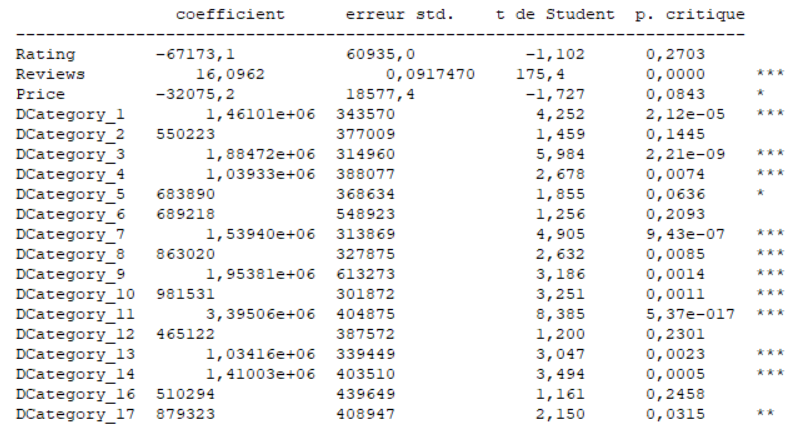


Qu’est-ce que ceci signifie ? Notre p-critique tout à droite est ce, sur quoi nous nous sommes basés pour notre méthode. En effet, nous pouvons remarquer qu’il n’y a plus de p-value supérieur à 5%. De plus, GRETL nous spécifie via un système d’étoile, si une variable indépendante est très prédictive ou non. Ici nous pouvons surtout remarquer que la p-value de Reviews est de 0. Nous avons donc ci-dessus, les variables indépendantes qui prédisent au mieux notre variable dépendante « Installs ».



*9.7 Mise en pratique – Dataset GAME*

Sans grand étonnement, nous allons ici procéder de la même façon que précédemment. Nous allons réutiliser la méthode du « BACKWARD – ELIMINATION ». Nous prenons à nouveau un seuil de 5 % et nous réutilisons les mêmes variables indépendantes.



Nous pouvons constater que nous avons beaucoup de variables indépendantes qui peuvent prédire notre variable dépendante. A nouveau nous pouvons constater que la variable le prédisant le mieux est Reviews qui a à nouveau obtenu une p-value de 0.

