|  |
| --- |
|  |

**RAPPORT DE PROJET 2I**

Analyse de traces d'un data center de Google

**Sommaire**

I) Introduction :

II) Méthodologie :

III) Analyse des données de la table «**job\_events** » :

1. Analyse de la caractéristique « ***event\_type*** »
2. Analyse de la caractéristique « ***time*** »
3. Analyse de la caractéristique « ***Scheduling\_class*** »
4. Analyse de la caractéristique « ***job\_ID*** »

IV) Analyse des données de la table « **task\_contraints** » :

1. Analyse de la caractéristique « ***comparaison\_operator*** »
2. Analyse de la caractéristique « ***job\_ID*** »
3. Analyse de la caractéristique « ***task\_index*** »
4. Analyse de la caractéristique « ***time*** »

V) Analyse des données de la table «**machine\_attributes** » :

1. Analyse de la caractéristique « ***time*** »
2. Analyse de la caractéristique « ***machine\_ID*** »
3. Analyse de la caractéristique « ***attribute deleted*** »

VI) Analyse des données de la table «**machine\_events** » :

1. Analyse de la caractéristique « ***machine\_ID*** »
2. Analyse de la caractéristique « ***time*** »
3. Analyse de la caractéristique « event ***type*** »

VII) Conclusion :

**Introduction**

Dans le cadre de notre deuxième année d’ingénieur en intelligence artificielle à l’ESME Sudria, nous avons fait un projet de 3 mois qui consiste à analyser un **data center** de **Google**. Pour cela nous avons téléchargé une base de données google et travaillé avec le langage **Python**. Nous avons aussi utilisé les librairies **Scikit-learn**, **Numpy** et **Pandas**. L’objectif de notre projet est d’analyser les données collectées et tirer des conclusions utiles sur les machines, les travaux et les tâches ainsi que l'utilisation des ressources dans le Data Center de Google. Les données collectées proviennent d’un data center de google sur une durée de 29 jours datant de 2011.

Ces données sont réparties sur six tables différentes, chacune d’elles recouvre un aspect des ces données. Ces tables sont : « ***job\_events*** », « ***machine\_attributes*** », « ***machine\_events*** », « ***task\_constraints***», « ***task\_events*** » et « ***task\_usage*** ».

Ce rapport présentera les différentes méthodes/outils utilisés pour les analyses. Et ensuite les analyses des tables « ***job\_events*** », « ***task\_constraints*** », « ***machine\_attributes*** » et « ***machine\_events*** » et de leurs colonnes.

Les tables « ***task\_events*** » et « ***task\_usage*** » ne seront pas analysés à cause de leur trop grande taille générant des erreurs de mémoires lors des tentatives pour les analyser.

**Méthodologie**

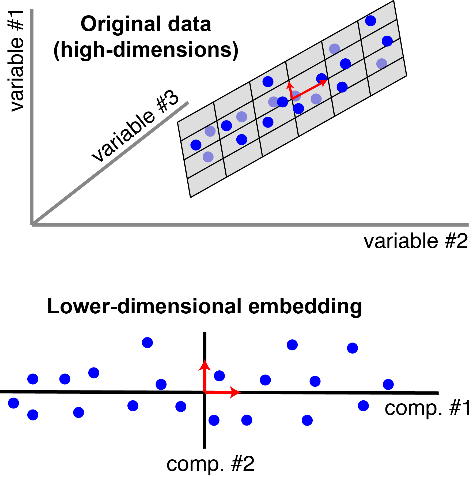
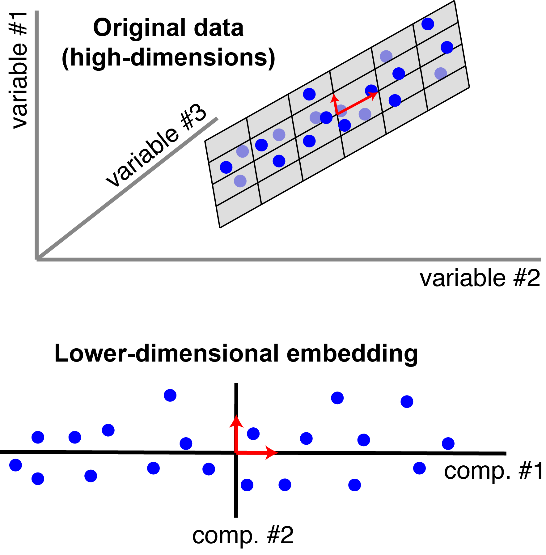
Dans notre projet, nous avons analysé des traces collectées à partir des données opérationnelles de Data Center de Google sur 29 jours. Cette analyse fut faite grâce à l’utilisation de plusieurs outils de machine learning implémenter dans des programmes **Python** et des librairies du même langage.

Les données du dataset sont récupérer et utilisé par les programmes grâce à la librairie **Pandas**, cette librairie est spécialisée notamment dans la création et la gestion de structure de données pouvant être utilisé notamment pour le machine learning.

Pour le machine learning la méthode utilisée est celle d’apprentissage supervisé **K-Means**, cet algorithme permet de déterminer dans un ensemble de données un nombre défini de groupes, clusters, dont le contenu est assez homogène et assez distinct des autres clusters. Pour ce faire, l’algorithme K-Means définit de manière aléatoire des centroïdes pour les clusters, puis calcule la distance moyenne entre une données et les centroïdes en utilisant les valeurs numériques de chaque caractéristiques, colonnes, de cette donnée. Et cette donnée est affecté au cluster dont la distance avec le centroïde est la plus faible. Après les centroïde de chaque cluster sont recalculés et ce processus est répété pour chacune des données. Pour ce faire il faut donc que les caractéristiques des données soient des valeurs numériques, c’est pourquoi nous choisirons de ne pas traiter les colonnes contenant des valeurs non numériques.

Une fois les données traitées avec K-Means, un autre outil est utilisé. Il s’agit du **Principal Component Analysis**, (**PCA**) issus de la librairie *sklearn.decomposition*. Cet outil permet de réduire la dimension d’une donnée en ne prenant en compte que les caractéristiques les plus importantes. Le nombre de ces caractéristiques peut être choisi. Le PCA est un outil souvent utilisé en machine learning dans les cas où les données possèdent un grand nombre de dimensions parce que cela permet de gagner beaucoup de ressources et de temps de calcul. En effet, plus le nombre de dimensions des données d’un dataset est élevé, plus le traitement du dataset sera long et couteux en ressources car chacune de ces dimensions seront prises en compte et utilisées pour les calculs. Même si toutes ces dimensions n’ont pas la même importance sur le résultat qui nous intéresse, que nous cherchons à prédire. Or dans certains cas, certaines composantes n’ont qu’un impact négligeable sur le résultat recherché. C’est pourquoi prendre en compte ces composantes n’est pas très utile et ne résulte finalement qu’à de la perte de temps. L’utilisation du PCA permet de régler ce problème et ainsi d’économiser du temps de calcul et des ressources, sans pour autant diminuer significativement la qualité de l’analyse qui suit.

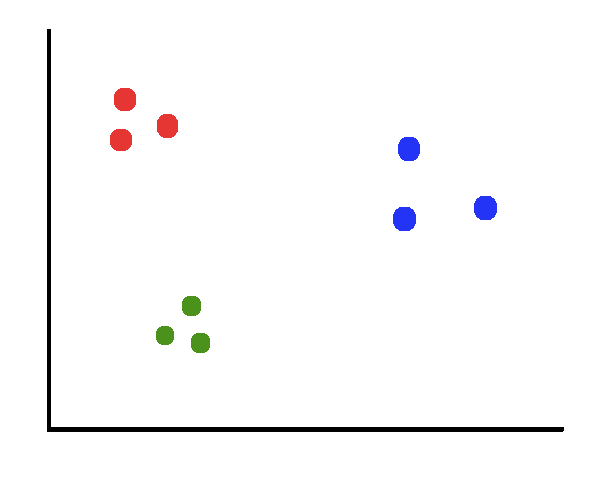
Voici un graphique illustrant le principe du PCA :

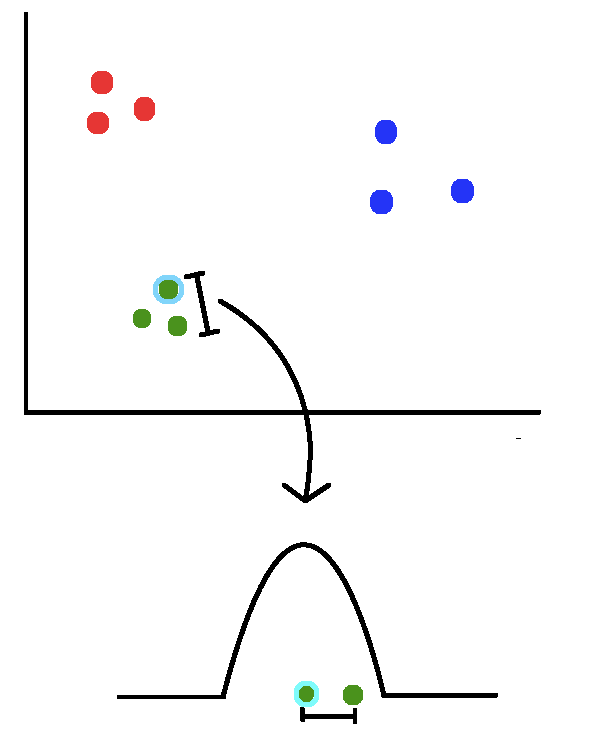


**PCA**

Dans notre solution, seuls les 3 composantes les plus importantes seront sélectionnées, excepté s’il y a moins de 3 composantes. Dans ce dernier cas, le PCA nous permet uniquement de les trier par ordre d’importance.

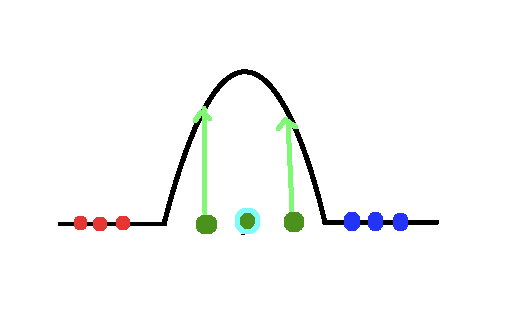
Un autre algorithme de réduction de dimension est utilisé, il s’agit du **t-SNE**, **t-distributed stochastic neighbor embedding**, issu de la librairie *sklearn.manifold*. Contrairement au PCA, il réduit le dataset à seulement deux ou trois dimensions, afin de permettre afficher le résultat directement. Cet algorithme regroupe les données en calculant la probabilité de similarité entre elle.

 Voici des graphiques illustrant le principe du t-SNE :



Voici l’ensemble de départ

Un point est pris comme référence pour le premier cluster.



Et les autres points sont répartie dans des clusters selon la probabilité selon leur probabilité de similarité.

Cependant les graphiques obtenus avec le t-SNE ne permettent pas d’enrichir nos analyses, donc ces graphes n’apparaitront pas dans la suite du rapport.

Une autre librairie Python a aussi été utilisée pour visualiser les résultats, il s’agit de la librairie **matplotlib.pyplot** de Python. Elle permet de faciliter la création de graphiques divers et variés, elle est très utilisée dans de nombreux domaines différents les uns des autres. Matplotlib.pyplot est une librairie dédiée à la création de graphiques et contient donc de nombreuses méthodes permettant d’y parvenir.

Analyse des données de la table «***job\_events*** »

La table «***job\_events*** » est constitué de 2 012 242 enregistrements des événements et des cycles de vie de 672 074 travaux. Ces enregistrements contiennent 8 caractéristiques : « *time* », «*missing info* », « *job ID* », « *event type* », « *user* », « *scheduling class* », « *job name* » et « *logical job name* ». Cependant nous n’étudierons pas les caractéristiques suivantes : « *missing info* », « *user* »,   
« *job name* » et « *logical job name* » car elles contiennent des valeurs non numériques.

Nous étudierons donc les caractéristiques « *time* », « *job ID* », « *event type* » et « *scheduling class* » afin de pouvoir observer ce qu’elles peuvent nous apprendre.

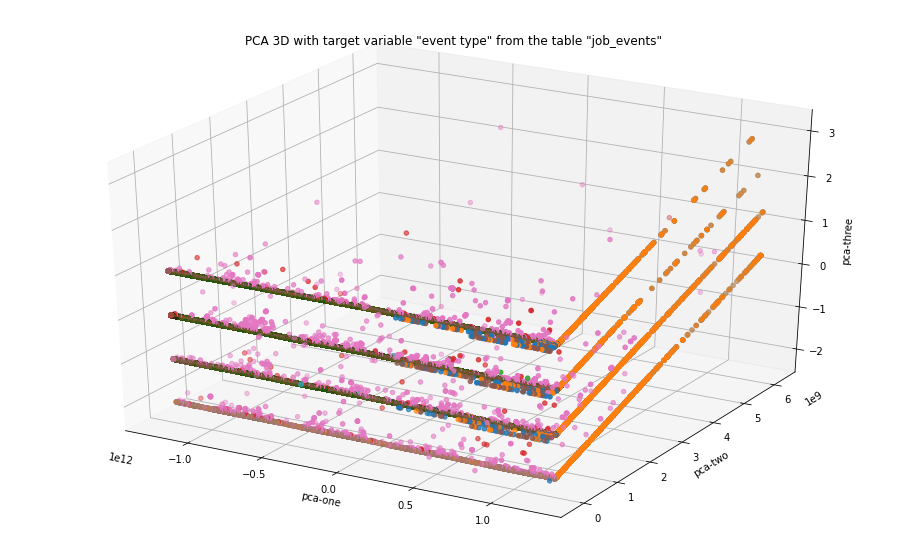
a) Analyse de la caractéristique « ***event\_type*** » :

La caractéristique « ***event\_type*** » représente l’identifiant de l’événement enregistré du travail, il s’agit d’un entier comprit entre 0et 8, chacun de ces nombres fait référence à un type d’événement précis. Nous nous y intéressons afin de savoir s’ils sont générés selon un motif particulier ou de façon totalement aléatoirement.

On observe que quatre groupes d’événement d’égale importance apparaissent sur le graphique du PCA 3D.

Et on peut déterminer grâce au graphique scatter de la table « *job\_events* » (cf. la fin du chapitre) que « *pca-one* », « *pca-two* » et « *pca-three* » correspondent respectivement aux colonnes « *job ID* », « *time* », et « *scheduling\_class* ».

On peut en conclure que les neuf événements peuvent être regroupés en quatre groupes de taille similaires.



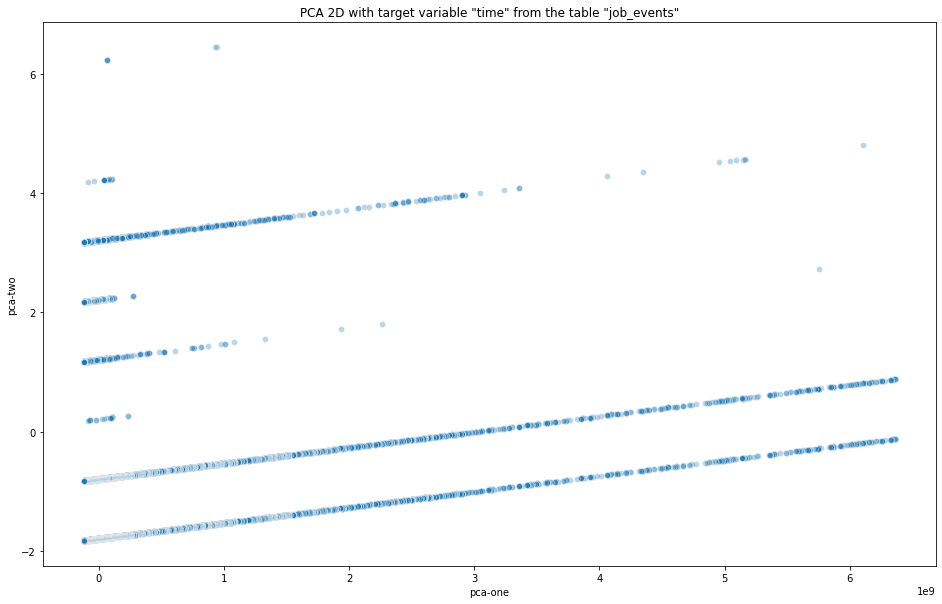
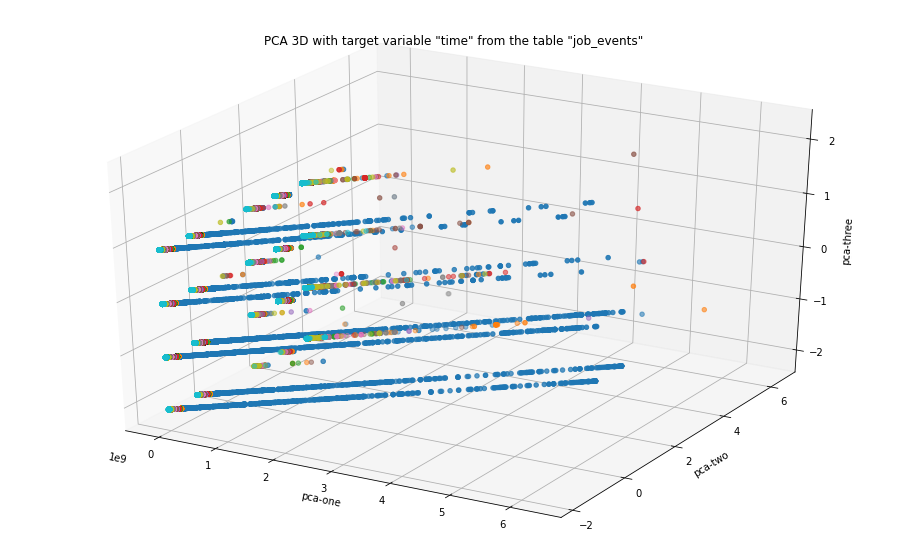
b) Analyse de la caractéristique « ***time*** » :

La colonne « ***time*** » représente le moment où l’événement du travail à été enregistré. Nous nous y intéressons pour savoir si ces événements apparaissent selon un motif particulier.

On peut remarquer grâce au PCA 2D que sept groupes distincts apparaissent. Nous pouvons déduire du PCA 3D que ces groupes semble se répéter quatre fois dans l’intervalle de temps durant lequel s’est effectué les enregistrements du dataset.

Et on peut déterminer grâce au graphique scatter de la table « *job\_events* » que « *pca-one* », « *pca-two* » et « *pca-three* » correspondent respectivement aux colonnes « *job\_ID* », « *event\_type* », et « *scheduling\_class* ».

On peut donc en conclure que les événements surviennent dans le temps selon un motif particulier.



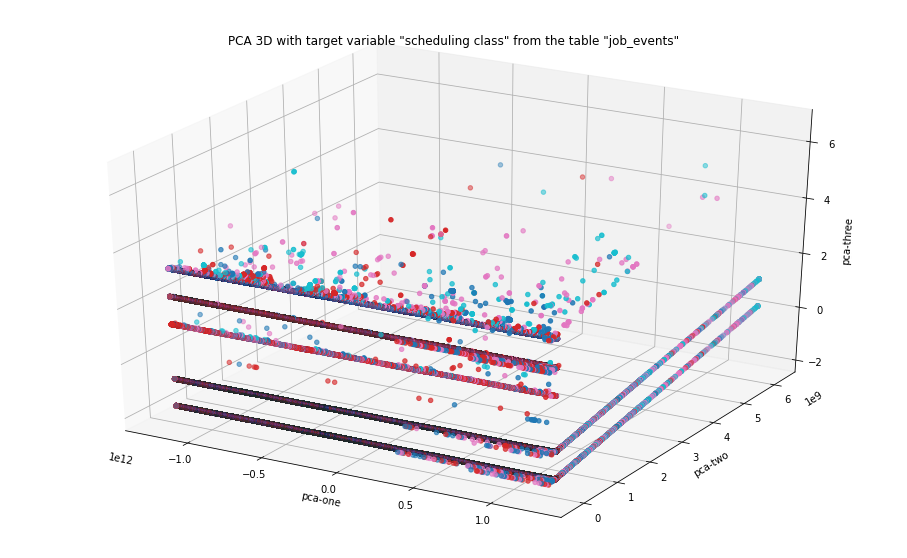
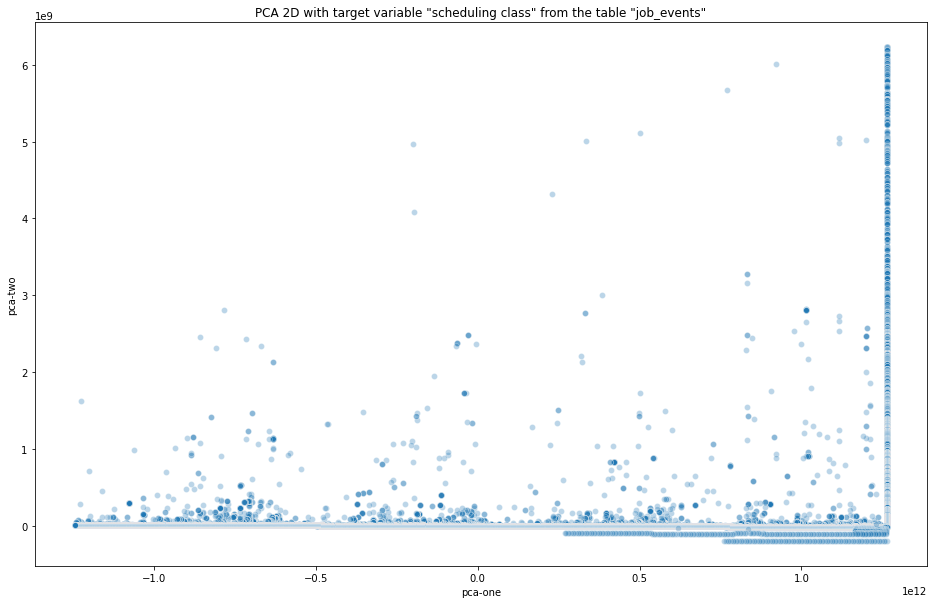
c) Analyse de la caractéristique « ***Scheduling\_class*** » :

La caractéristique « ***Scheduling\_class*** » représente la sensibilité à la latence du travail, autrement dit l’importance de l’impacte négatif que peut avoir la latence pour le travail. Cette sensibilité à la latence ne doit pas être confondu avec la priorité du travail, même si cela peut être relativement corrélé. Nous nous y intéressons afin de savoir si les travaux sont répartis uniformément selon cette sensibilité à la latence ou si au contraire certaines sensibilités apparaissent en plus grand nombre. Nous souhaitons aussi déterminer si cette répartition est continue ou discrète.

On observe grâce au PCA 3D qu’ils peuvent être séparés assez distinctement en cinq groupes selon l’importance de la latence. On peut aussi noter que deux de ces groupes apparaissent nettement plus important en nombre comme le souligne la figure du PCA 2D.

Et on peut déterminer grâce au graphique scatter de la table « *job\_events* » que « *pca-one* », « *pca-two* » et « *pca-three* » correspondent respectivement aux colonnes « *time* », « *job\_ID* », et « *event\_type* ».

On peut donc en conclure que les travaux sont classés par les datacenters selon leur sensibilité à la latence.

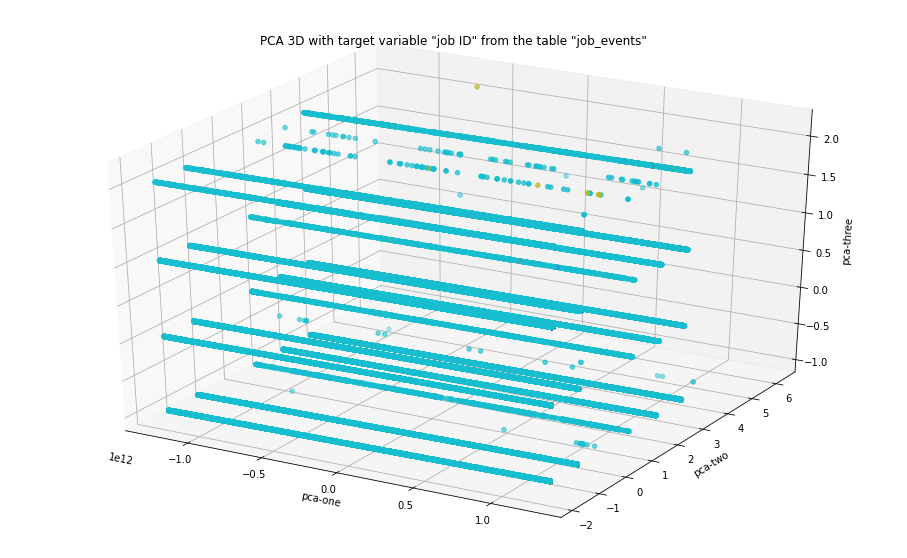


d) Analyse de la caractéristique « ***job\_ID*** » :

La caractéristique « ***job\_ID*** » représente l’identifiant du travail, il s’agit d’un nombre généré automatiquement par le data center. Nous nous y intéressons afin de savoir s’ils sont générés selon un motif particulier ou de façon totalement aléatoirement.

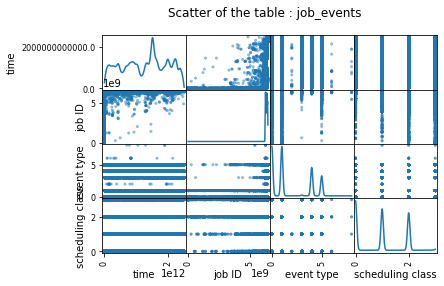
On peut remarquer grâce au PCA 2D que cinq gros groupes distincts apparaissent auxquels s’ajoutent trois autres groupes restreints. Les cinq premiers groupes sont relativement aussi important les uns par rapport aux autres. Le PCA 3D nous permet d’observer que ces huit groupes se répètent quatre fois.

Et on peut déterminer grâce au graphique scatter de la table « *job\_events* » que « *pca-one* », « *pca-two* » et « *pca-three* » correspondent respectivement aux colonnes « *time* », « *event\_type* », et « *scheduling\_class* ».

 On peut donc en conclure que les identifiants des suivent un motif bien particulier.

Une image contenant table

Description générée automatiquement



Analyse des données de la table «***task\_constraints*** »

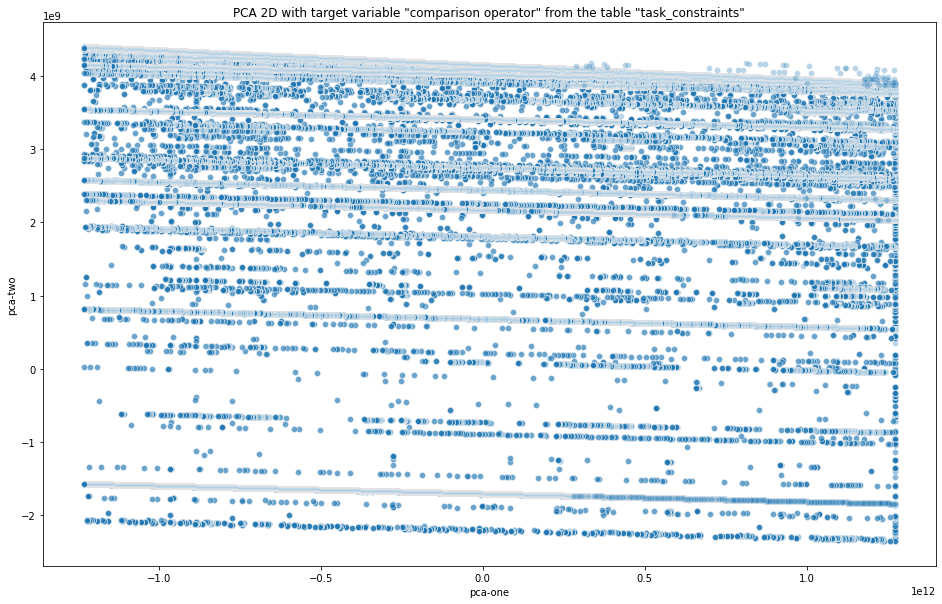
Cette table contient des enregistrements des tâches effectué par les machines des data center de Google. Et plus précisément, elle contient les contraintes que ces tâches imposent aux machines, chacune de ces tâches n’ont pas forcément le même nombre de contraintes.

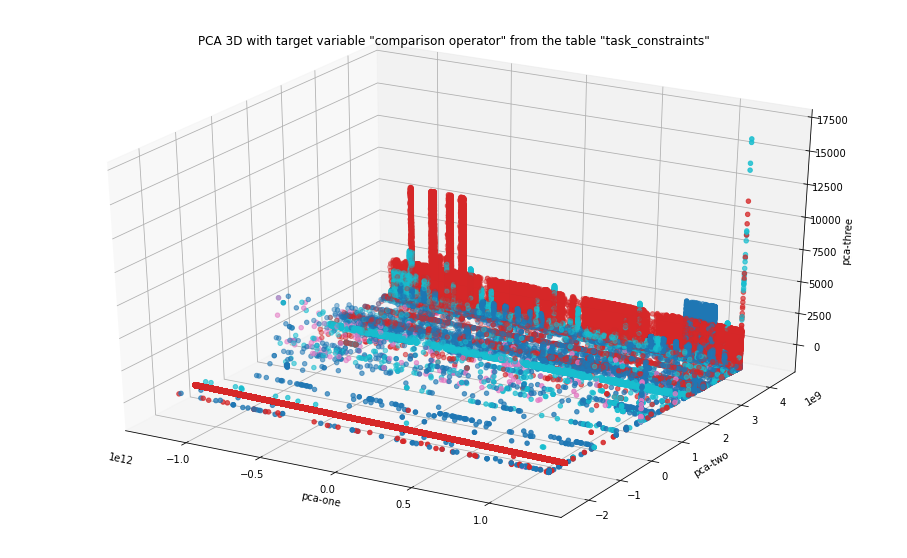
a) Analyse de la caractéristique « ***Comparaison\_operator*** » :

La caractéristique « *comparaison operator* » indique la manière dont l’attribut d’une machine doit être représenter et ainsi comment elle peut être comparé. L’attribut peut être représenter comme une chaine de caractére ou comme un entier.

On remarque sur le PCA 2D un nombre important de points sans regroupement nette.

Sur le PCA 3D, si on retrouve aussi cet amas de points, on peut tout de même distinguer qu’il n’y a qu’une seule couche. Sur cette unique couche, on remarque 4 groupes identiques et assez proche ainsi qu’un cinquième plus gros.

Et on peut déterminer grâce au graphique scatter de la table « *task\_contraints* » que « *pca-one* », « *pca-two* » et « *pca-three* » correspondent respectivement aux colonnes « *time* », «*job\_id* », et « *task\_index* ».



b) Analyse de la caractéristique « ***job\_ID*** » :

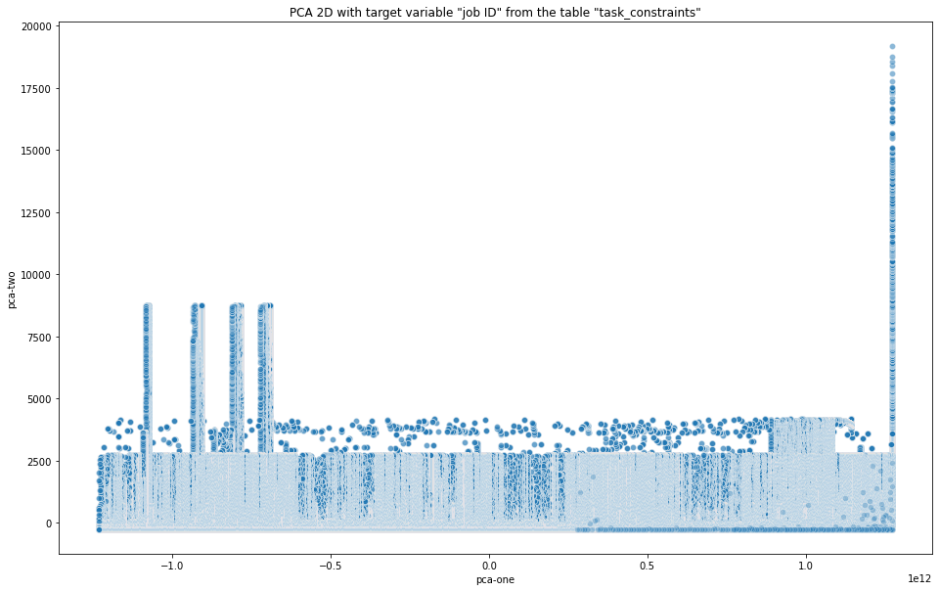
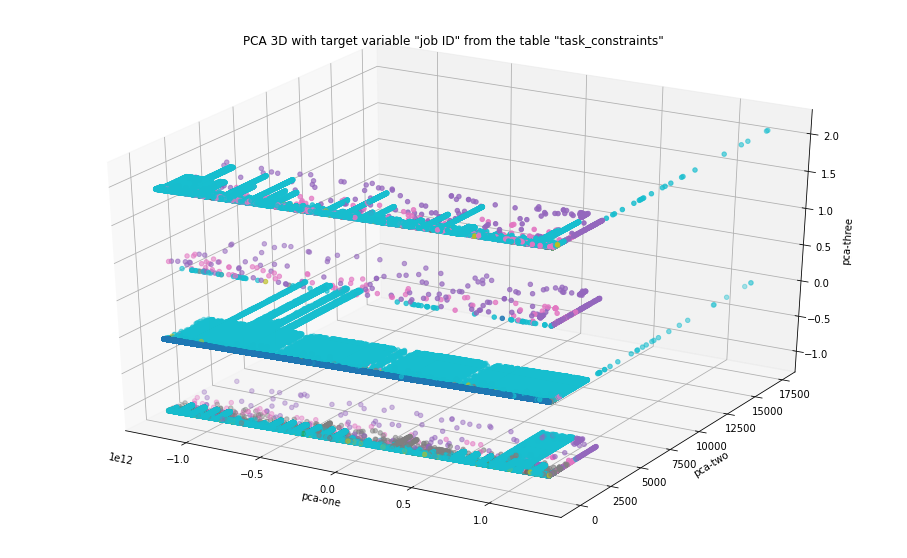
Tout comme pour « **Job\_events** »,la caractéristique « ***job\_ID*** » de *«****task\_contraints*** », représente l’identifiant du travail, et est un nombre généré automatiquement par le data center.

On peut remarquer grâce au PCA 2D qu’il y a 4 groupes de valeurs identiques ainsi qu’un cinquième plus gros que les autres. On peut aussi remarquer qu’il existe un gros groupe formant une « base » présent pour toutes les valeurs.

On observe grâce au PCA 3D que les travaux peuvent être séparés assez distinctement en quatre groupes. On retrouve aussi la « base » ainsi que les 4 autres groupes.

On peut déterminer grâce au graphique scatter de la table « *task\_contraints* » que « *pca-one* », « *pca-two* » et « *pca-three* » correspondent respectivement aux colonnes « *time* », «*task\_index* », et « *comparaison\_operator* ».

On peut donc en conclure que pour « task\_contraints » les identifiants des travaux suivent aussi un motif précis.



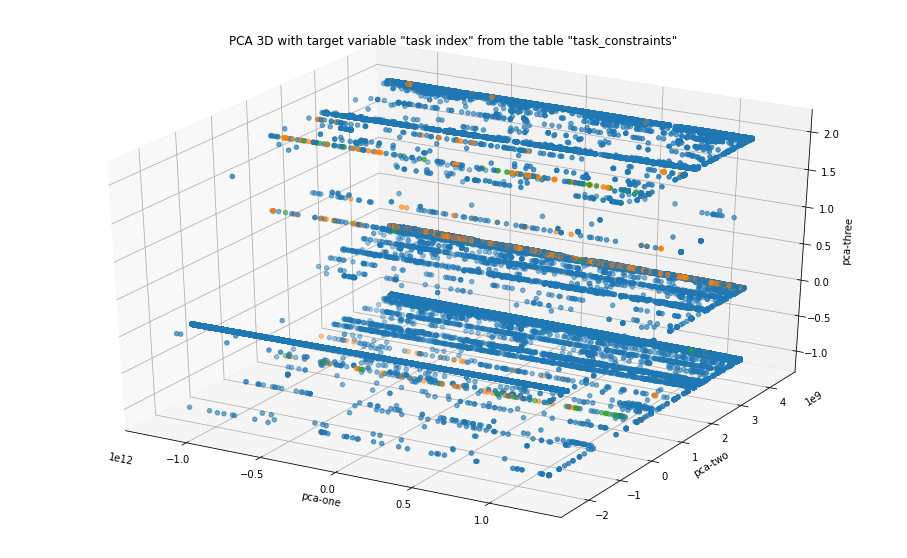
c) Analyse de la caractéristique « ***task\_index*** » :

Pour « task\_index » le PCA 2D ne nous apprend pas grand-chose mis à part que les points n’ont pas l’air de suivre un schéma particulier.

En revanche, sur le PCA 3D on peut distinguer 4 couches distinctes. On peut donc en déduire que si les points apparaissent de manière aléatoire on peut les regrouper en 4 groupes distincts.

Une image contenant texte

Description générée automatiquementOn peut déterminer grâce au graphique scatter de la table « *task\_contraints* » que « *pca-one* », « *pca-two* » et « *pca-three* » correspondent respectivement aux colonnes « *time* », «*job\_id* », et « *comparaison\_operator* ».

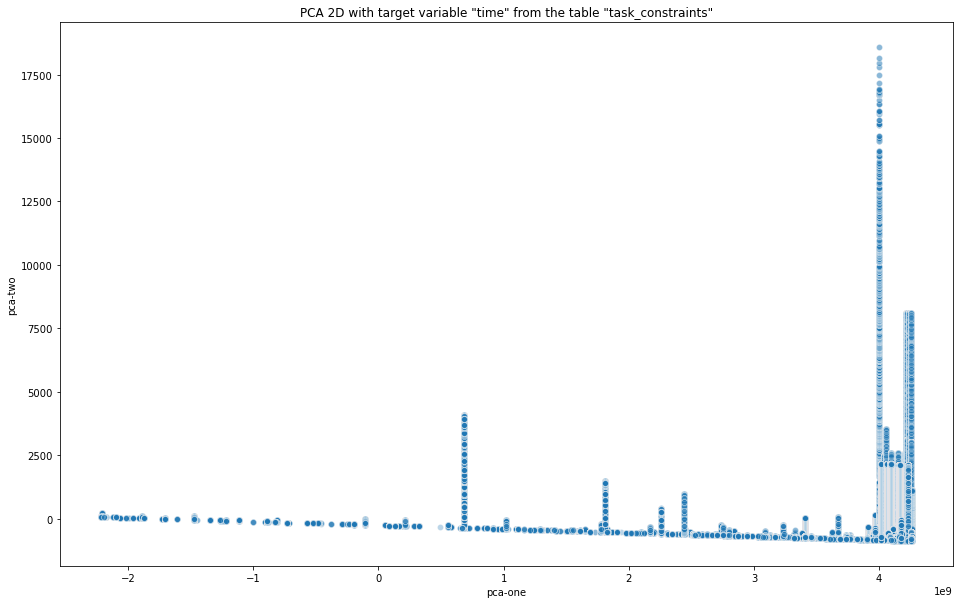


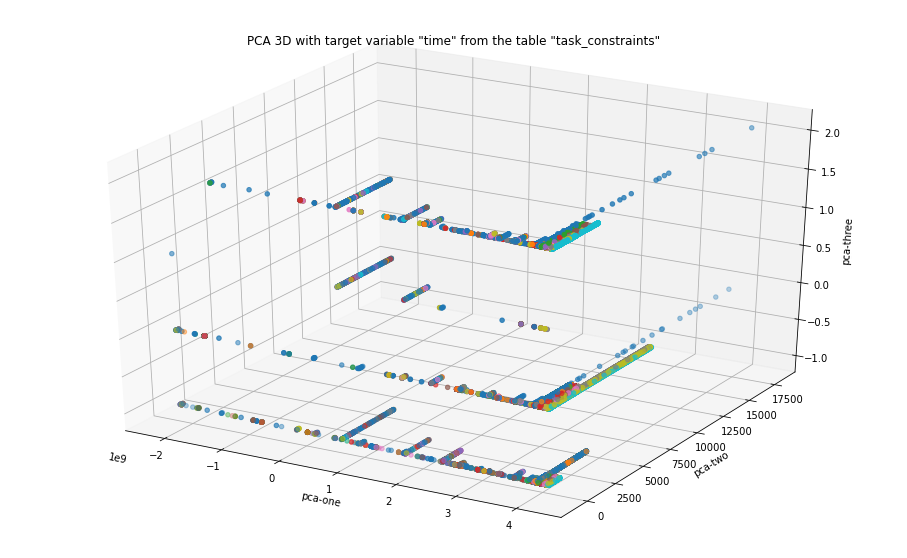
d) Analyse de la caractéristique « ***time*** » :

La colonne « ***time*** » représente le moment où l’état de la machine à été enregistré. Nous nous y intéressons pour savoir si ces tâches sont effectués selon un motif particulier.

Sur le PCA 2D on observe six groupes distincts qui apparaissent. Nous pouvons déduire du PCA 3D que ces groupes semble se répéter quatre fois dans l’intervalle de temps durant lequel s’est effectué les enregistrements du dataset.

On peut déterminer grâce au graphique scatter de la table « *task\_contraints* » que « *pca-one* », « *pca-two* » et « *pca-three* » correspondent respectivement aux colonnes « *job\_ID* », «*task\_index* », et « *comparaison\_operator* ».

On peut donc en conclure que les événements surviennent dans le temps selon un motif particulier.

Analyse des données de la table «***machine\_attributes*** »

La table «***machine\_attributes*** » est constitué de 10 748 566 enregistrements des propriétés de 12 583 machines des data center de Google. Ces enregistrements contiennent 5 caractéristiques : « *time* », «*machine ID* », « *attribute name* », « *attribute value* » et « *attribute deleted* ». Cependant nous n’étudierons pas les caractéristiques suivantes : « *attribute name* » et « *attribute value* » car elles contiennent comme valeurs des chaines de caractères.

Nous étudierons donc les caractéristiques « *time* », « *machine ID* » et « *attribute deleted* » afin de pouvoir observer ce qu’elles peuvent nous apprendre.

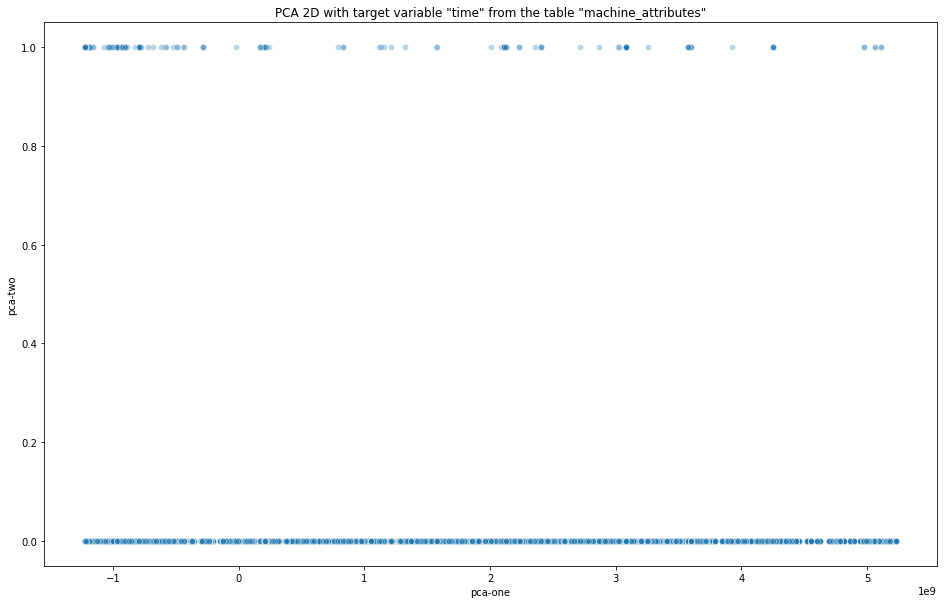
a) Analyse de la caractéristique « ***time*** » :

La colonne « ***time*** » représente le moment où l’état de la machine à été enregistré. Nous nous y intéressons pour savoir si un motif particulier apparait.

On peut remarquer sur le PCA 2D que deux lignes horizontales apparaissent en bas et en haut du graphique. Et que celle du bas est nettement plus importante que celle du haut.

Et on peut déterminer grâce au graphique scatter de la table « *machine\_attributes* » que « *pca-one* » et « *pca-two* » correspondent respectivement aux colonnes « *machine ID* » et   
« *attribute deleted* ».

On peut donc en conclure que les événements surviennent dans le temps selon un motif très particulier regroupant les machines en deux catégories distinctes avec des caractéristique qui doivent être très spécifique.



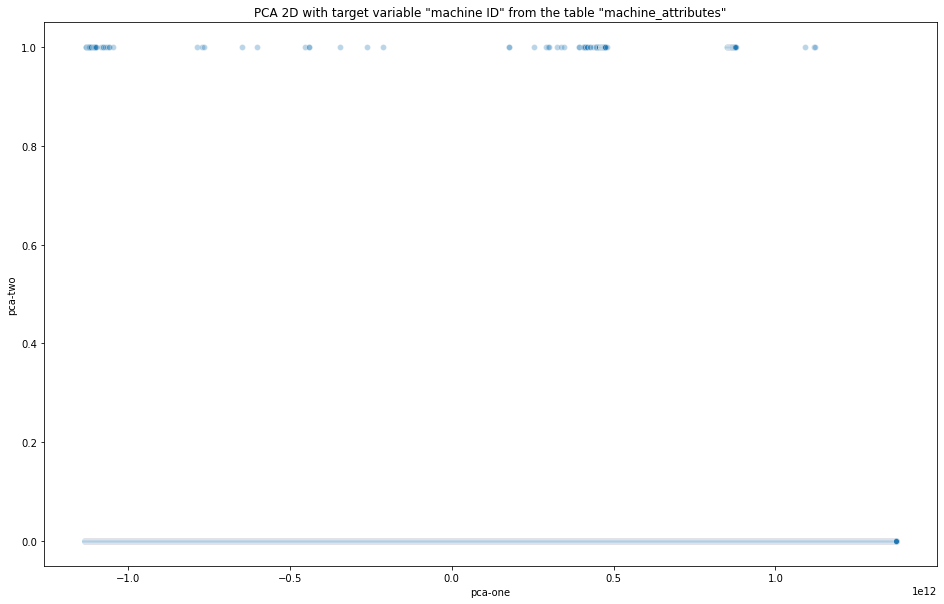
b) Analyse de la caractéristique « ***machine\_ID*** » :

La caractéristique « ***machine\_ID*** » représente l’identifiant lié à la machine, il s’agit d’un nombre généré automatiquement par le data center. Nous nous y intéressons afin de savoir s’ils sont générés selon un motif particulier ou de façon totalement aléatoirement.

On peut remarquer grâce au PCA 2D un motif similaire que celui de la colonne « ***time*** », cependant les lignes sont moins denses.

Et on peut déterminer grâce au graphique scatter de la table « *machine\_attributes* » que « *pca-one* » et « *pca-two* » correspondent respectivement aux colonnes « *time* » et « *attribute deleted* ».

On peut donc en conclure que les identifiants des machines suivent un motif assez similaire à celui de lié au moment où les enregistrements sont effectués, ce qui indique que leur génération n’est pas aléatoire et est probablement lié au temps.



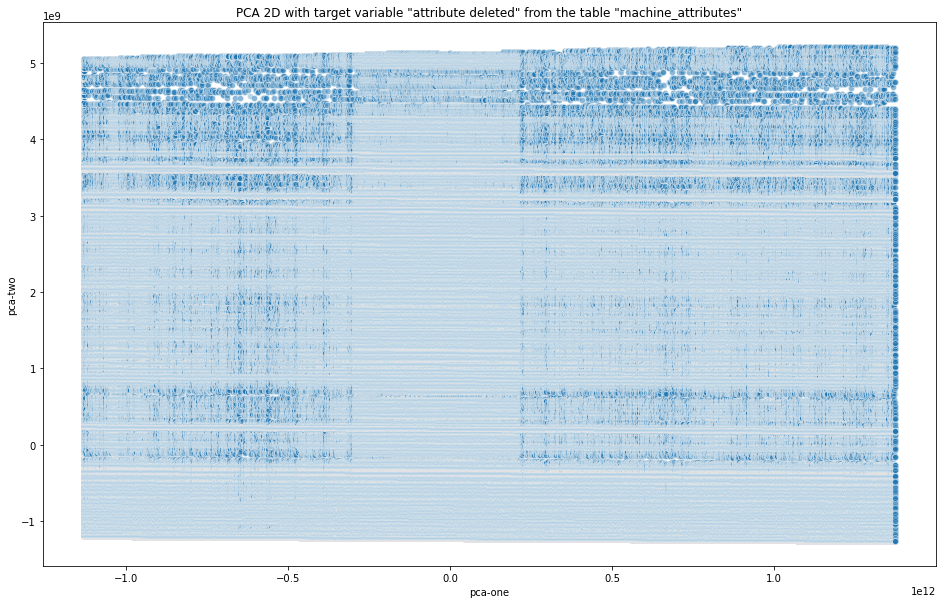
c) Analyse de la caractéristique « ***attribute deleted*** » :

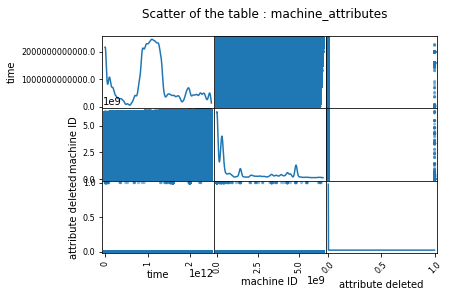
La caractéristique « ***attribute deleted*** » indique si l’attribut a été supprimé, elle est représentée par une valeur booléenne, donc égale soit à 0 (non supprimé) soit à 1 (supprimé). Nous nous y intéressons afin de savoir si les attributs sont uniformément supprimés.

Sur le PCA 2D un seul groupe semble se former, et celui-ci recouvre quasiment la totalité du graphique.

Et on peut déterminer grâce au graphique scatter de la table « *machine\_attributes* » que « *pca-one* » et « *pca-two* » correspondent respectivement aux colonnes « *machine ID* » et « *time* ».

On peut en conclure que quasiment la totalité des attributs de l’ensemble des machines sont regroupé dans une seule catégorie. Lorsque l’on regarde la quantité de 0 et celle de 1 dans la colonne « *attribute deleted* » de la table « *machine\_attributes* », on constate que cette catégorie est celle des attributs non supprimés.





Analyse des données de la table «***machine\_events*** »

La table «***machine\_events*** » est constitué de 10 748 566 enregistrements des événements de 12 583 machines des data center de Google. Ces enregistrements contiennent 5 caractéristiques : « *time* », «*machine ID* », « *event type* », « *platform ID* », « *CPUs* » et « *Memory* ». Cependant nous n’étudierons pas les caractéristiques suivantes : «*platform ID* », « *CPUs* » et « *Memory* » car elles contiennent comme valeurs non numériques.

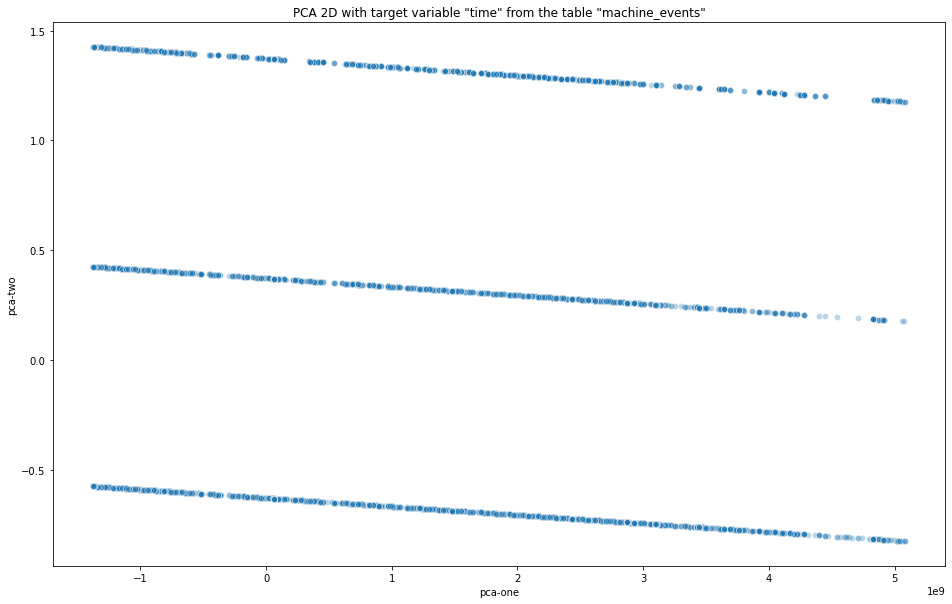
Nous étudierons donc les caractéristiques « *time* », « *machine ID* » et « *event type* » afin de pouvoir observer ce qu’elles peuvent nous apprendre.

a) Analyse de la caractéristique « ***time*** » :

La colonne « ***time*** » représente le moment où l’événements de la machine à été enregistré. Nous nous y intéressons pour savoir si un motif particulier apparait.

On peut remarquer sur le PCA 2D que trois ligne obliques se tracent sur le graphique, ces lignes semble être parallèle. On peu aussi noter que celle en bas est celle qui contient plus de point, et à l’inverse, celle en haut est celle qui en contient le moins.

Et on peut déterminer grâce au graphique scatter de la table « *machine\_events* » que « *pca-one* » et « *pca-two* » correspondent respectivement aux colonnes « *machine ID* » et « *event type* ».

 On peut donc en conclure que les événements surviennent dans le temps selon un motif très particulier regroupant les machines en deux catégories distinctes avec des caractéristique qui doivent être très spécifique.

b) Analyse de la caractéristique « ***machine\_ID*** » :

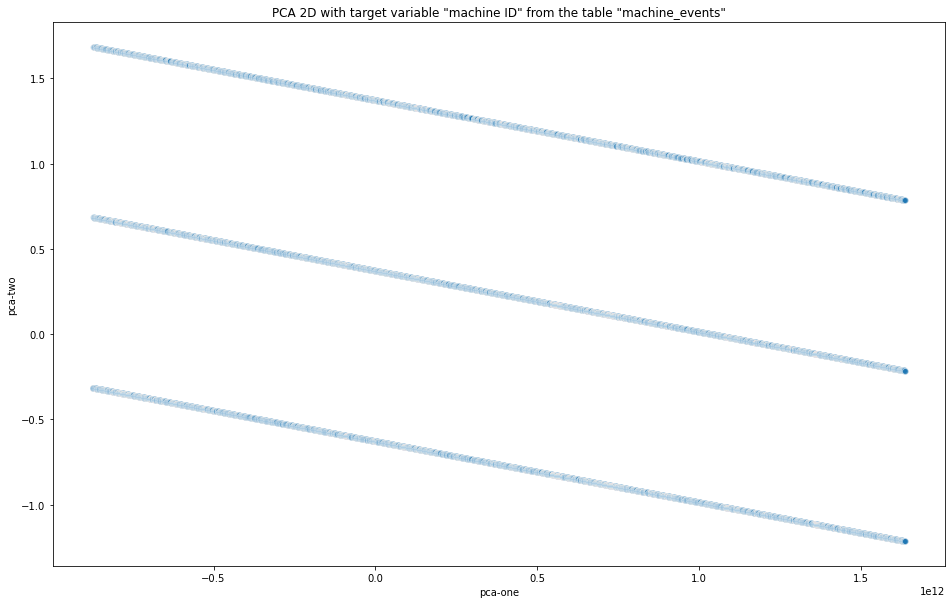
La caractéristique « ***machine\_ID*** » représente l’identifiant lié à la machine, il s’agit d’un nombre généré automatiquement par le data center. Nous nous y intéressons afin de savoir s’ils sont générés selon un motif particulier ou de façon totalement aléatoirement.

On peut remarquer grâce au PCA 2D un motif similaire que celui de la colonne « ***time*** », cependant les lignes sont moins denses.

Et on peut déterminer grâce au graphique scatter de la table « *machine\_events* » que « *pca-one* » et « *pca-two* » correspondent respectivement aux colonnes « *time* » et « *event type* ».

En analysant les identifiants contenus dans la colonne « ***machine\_ID*** » de la table   
«***machine\_events*** » et ceux de la table «***machine\_attributes*** », on remarque que ce sont les mêmes.

On peut donc en conclure que les identifiants des machines suivent un motif assez similaire à celui de lié au moment où les enregistrements sont effectués, ce qui indique que leur génération n’est pas aléatoire et est probablement lié au temps. Ce qui consolide la conclusion faites lors de l’analyse de la table «***machine\_attributes*** ».



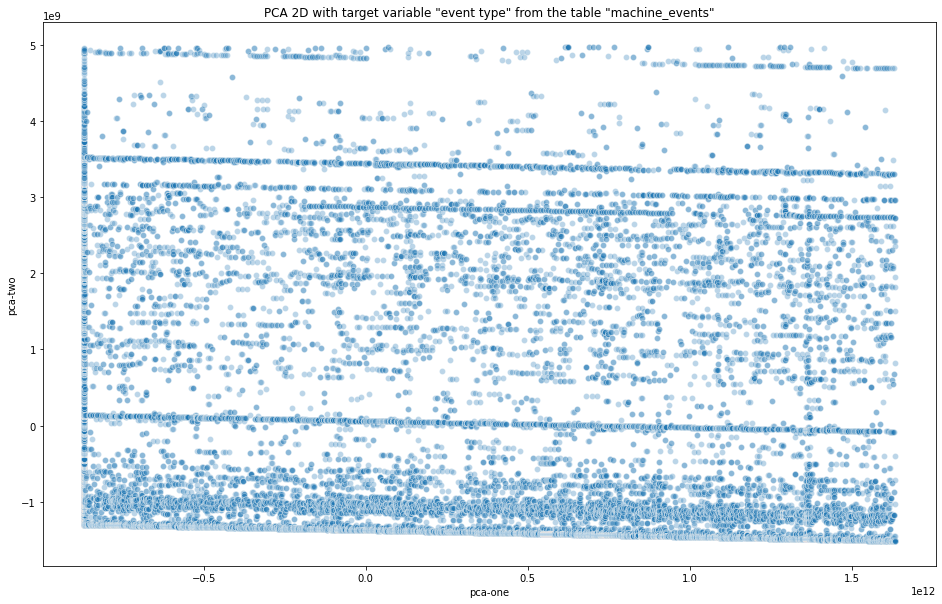
c) Analyse de la caractéristique « ***event type*** » :

La caractéristique « ***event type*** » indique le type d’événement lié à la machine. Il s’agit d’un entier comprit entre 0 et 2. 0 correspond à l’ajout d’une machine, chaque machine possède au moins une fois cet événement. Un 1 est associé au retrait de la machine, dû à un arrêt de fonctionnement ou à une maintenance. Et l’entier 3 indique la mise à jour des ressources de la machine. Nous nous y intéressons afin de savoir si des groupes d’événements apparaissent assez distinctement.

Sur le PCA 2D on peut observer un très grand nombre de points qui semble se répartir sur des lignes légèrement oblique dont la densité et le nombre s’accroit en descendant sur le graphique. A ces lignes obliques se rajoute une ligne verticale à l’extrême gauche du graphique.

Et on peut déterminer grâce au graphique scatter de la table « *machine\_events* » que « *pca-one* » et « *pca-two* » correspondent respectivement aux colonnes « *machine ID* » et « *time* ».

De ce graphique PCA 2D, on ne peut en conclure que les événements sont relativement distincts les uns des autres, même si une certaine dynamique peut être observé, aucun groupe clair ne se dessine clairement.



**Conclusion**

Grâce aux différents outils **Python** utilisés (**Scikit-learn**, **Numpy** et **Pandas**). Une analyse des tables « ***job\_events*** », « ***task\_constraints*** », « ***machine\_attributes*** » et « ***machine\_events*** » de l’ensemble de données à put être réaliser. Ces analyses ont permis d’avoir une meilleure appréhension sur les machines, les travaux et les tâches dans un des data centers de Google.

**Annexe**

Voici le lien Github vers le répertoire où est stocké le code utilisé pour ce projet :

<https://github.com/abosch38/projet_data>.