Supplementary materials of QuickInsights

Property of *impact*

We restrict the impact-measures to be measures only containing non-negative values. Paper [1] provides a set of calculations to accommodate anti-monotonic condition being held by various aggregations (e.g., top-k average for AVG). The corresponding calculations are denoted as $MonoAggr_i(insight.subspace)$. thus, we define impact by the following two steps:

Definition 1. $impact_i = \frac{MonoAggr_i(insight.subspace)}{MonoAggr_i(\{*\})}$

Definition 2. $impact = \max_i(impact_i)$

Lemma 1. Definition 2 satisfies anti-monotonic condition, and is bound between 0 and 1.

Proof: Considering the impact-measures are restricted to only contains non-negative values. since $MonoAggr_i(insight.subspace)$ satisfies anti-monotonic condition, thus,

$MonoAggr_i(insight.subspace) \leq MonoAggr_i(\{*\}) \rightarrow impact_i \in [0,1] \rightarrow impact \in [0,1]$, because every subspace is a subset of overall subspace {*}.

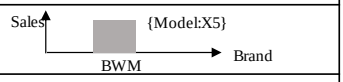
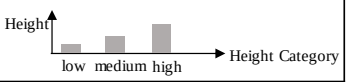
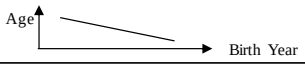
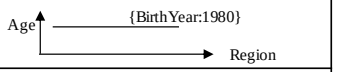
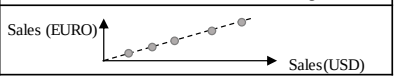
To proof anti-monotonic condition, let *S* and *s* be two subspaces where $s \in S$, then

$$impact_i(s) \leq impact_i(S)\ \forall i \rightarrow \max_i\big(impact_i(s)\big) \leq \max_i\big(impact_i(S)\big)$$

Functional-Dependency (FD) induced *trivial insights*

Definition 3 (functional dependency). A functional dependency FD: X → Y means that the values of Y are determined by the values of X, where X and Y are two sets of columns (i.e., dimensions or measures).

Table 1. Taxonomy of trivial insights

ID	Form of Functional-Dependency	Trivial insight description	Example
ID1	$\{s_1, \dots, s_p\} \rightarrow d$	Only one item in sibling group	
ID2	$\{m_1, \dots, m_q\} \rightarrow d$	Fixed x-y axis relationship	
ID3	$d \rightarrow \{m_1, \dots, m_q\}$	Fixed x-y axis relationship	
ID4	$\{s_1, \dots, s_p\} \rightarrow \{m_1, \dots, m_q\}$	Flat line	
ID5	$\{m_1, \dots, m_i\} \rightarrow m_j$	Fixed x-y axis relationship	

Definition 4 (FD of insight subject). We pick all the columns appearing in an insight subject as $Col := \{s_1, \dots, s_p, d, m_1, \dots, m_q\}$, where $s_1 \sim s_p$ are the dimensions appearing in subspace(s), d is breakdown, and

$m_1 \sim m_q$ are q measures. If $\exists\, X \subset Col, Y \subset Col, X \cap Y = \emptyset, s.t. X \to Y$, we say $X \to Y$ is a FD of this insight subject.

Based on Definition 6, we notice that FD of insight subject would bring up trivial insights.

Definition 5 (trivial insight). An insight is trivial if its aggregated values exhibit pre-determined relationships thus providing trivially useful information for the purpose of data analysis.

We carefully inspect all possible FDs incurred in insight subject, and come up to five forms of FD that would induce trivial insights, as shown in Table 1 (ID1 ~ ID5).

Explication:

Part I: we explicate the validity of the proposed 5 forms ID1~ID5.

We first check the cases where the insight subject contains single subspace and single measure. For exceptional insight types, 2DClustering (1 subspace, 2 measures), Cross-Measure Correlation (1 subspace, 2 measures), Correlation (2 subspaces, 1 measure), we check them later.

ID1: since the breakdown dimension *d* can be determined by subspace {s1~sp}, thus the generated sibling group only contains one item, which is with same aggregation value compared to the aggregation value of subspace {s1~sp}, thus is pre-determined.

ID2: the breakdown dimension *d* can be determined by measure *m*. In general, most real cases of ID2 are about the value of *d* is categorization or discretization of measure *m*. thus, the aggregation values after breakdown exhibits monotonicity, hence which is pre-determined. For example, the value of *Height-Category* is calculated by measure *height*, by setting $low = height \leq 100, high = height \geq 1000, medium = 100 < height < 1000.$ Any insight describes *height* breakdown by *Height-Category* would become a trivial *Outstanding No. 1* insight: "*height* of *high* is *outstanding No. 1* among all *Height-Categories*", which is pre-determined no matter what subspace of the insight is.

ID3: measure *m* can be determined by breakdown dimension *d*. draw a curve with x-axis values are the values of dimension *d*, and the y-axis values are the corresponding aggregation values of measure *m*. it is obviously this is a pre-fixed curve. For example, suppose there's a dimension BirthYear, and a measure Age, so BirthYear can determine Age, the value of Age is simply the current year minus the year of birth. It is very obvious that the earlier the birthday, the older the age, which is totally irrelevant to the insight subspace.

ID4: subspace {s1~sp} determines measure *m*. in this case, given the subspace of the insight subject, the aggregation value of measure is fixed, thus the aggregation results eventually is a flat line if drawn by a visual. For example, if the subspace if <BirthYear:1990>, then the Age = 27 no mater breakdown by any dimension such as Region or Gender.

ID5: ID5 refers to the insights with multiple measures. currently, only 2DClustering and Cross-Measure Correlation insights are with two measures. thus, we simplify ID5 as two measures, m1 determines m2. Draw a curve with x-axis values are the values of m1, and y-axis values are the corresponding values of m2, again, this curve is pre-fixed. For example, suppose there exists FD between two measures $Sales(USD) \to Sales(EURO)$ (falls into the category of ID5 in Table 1), the corresponding values only differ by a constant exchange-rate. These two measures will exhibit perfect correlation no matter breakdown by

any dimension (thus the relationship is pre-determined) when drawn in a scatter plot, but clearly provides trivial value for analysis.

Addition: we only left with the case of insight with multiple subspaces, which is the Correlation insight. Let's recap ID1 and ID4, suppose any one subspace (denoted as s1) of a correlation insight can determine the measure or breakdown dimension, thus the corresponding curve of s1 is pre-fixed, and it makes no sense to further evaluate the correlation between the curve of s1 and the curve of another subspace.

Part II: besides there ID1~ID5 forms, there are three other forms:

1) $\{s_1, \dots s_i\} \rightarrow s_j$: part of the subspace determines some other part
2) $d \rightarrow s_i$: breakdown determines part of the subspace
3) $\{m_1, \dots m_i\} \rightarrow \{s_1, \dots s_p\}$: measures determine the subspace

Definition 6 (duplicate subspaces): given 2 subspaces indicated by $\{s_1, \dots s_p\}$, if $\{s'_1, \dots s'_p\}$, we say the they are duplicate subspaces *iff* $\{s_1, \dots s_p\} \rightarrow \{s'_1, \dots s'_p\}$ and $\{s'_1, \dots s'_p\} \rightarrow \{s_1, \dots s_p\}$. For example, {BMW}{X5} and {X5} are duplicate subspaces considering dimension "Model" determines dimension "Brand".

Lemma 2: duplicate subspaces form a set of equivalent classes.

Proof: if subspace A is duplicate to subspace B, then B is duplicate to A (symmetry); if A is duplicate to B, and B is duplicate to C, then A is duplicate to C (transitivity); A is duplicate to A (reflexivity).

Definition 7 (concise subspace): given a set of subspaces within one equivalent class, a subspace is called *concise* if it is not a child of any other subspace.

These three forms don't induce trivial insights, but 1) and 2) would induce duplicate subspaces, so which will be used in the module of Subject-Searcher; while 3) is a case with no harm. Specifically,

1) implies that $\{s_1, \dots s_p\}$ and $\{s_1, \dots s_{j-1}, s_{j+1}, \dots s_p\}$ are duplicate subspaces. Thus, in the Subject-Searcher module, we only need to use only one of them for insight evaluation and discard the other one. In our implementation, within an equivalent class of duplicate subspaces, we pick the *concise* subspace for evaluation.

2) implies that $\{s_1, \dots s_p, d\}$ and $\{s_1, \dots s_{i-1}, s_{i+1} \dots s_p, d\}$ are duplicate subspaces, we should discard the former one in Subject-Searcher module.

3) is a form with no harm. Although the measure has certain determination on the subspace, the curve generated by breakdown is still unknown. For example, suppose the measure is *height*, and now the subspace is "Height-Category=High", then by breakdown of year, there still would exist seasonality/trend/outlier, we don't know. Thus, the curve is not pre-determined.

Time complexity of FD checker

Given an insight candidate, we need to check if there exists FDs to satisfy any of ID1~5 in Table 1 thus to avoid further insight evaluation. The checking can be generalized as determining whether $\{d_1 \sim d_i\} \rightarrow d_j$ is held or not. On the other hand, such determination requires knowing the FDs that are globally held in

a given dataset, and such FDs can be obtained from data schema or can be pre-calculated using FD mining techniques such as [2]. Thus, we formulate the problem as:

Problem (FD checker). Given a set of FDs $\{X_1 \rightarrow Y_1\}, \ldots, \{X_t \rightarrow Y_t\}$, check if $\{d_1 \sim d_i\} \rightarrow d_j$ is held or not.

This problem can be solved by leveraging two axioms in the field of FD theory: *Reflexivity* and *Transitivity* [3]. Roughly, if $d_j \in \{d_1 \sim d_i\}$, the $\{d_1 \sim d_i\} \rightarrow d_j$ is true (Reflexivity). Otherwise, find $X = \cup_i \{X_i | d_j \in Y_i\}$, and check if $\{d_1 \sim d_i\} \rightarrow X$ (Transitivity). This process repeats recursively until an empty set is reached.

Table 2. Pseudo code of FD checking

```
/* check if a set of columns determine another column, given a set of
   basicFDs */
IsDependent(determinantCols, col, basicFDs)
   inspected ← {}
   return Qualify(determinantCols, col, inspected, basicFDs)

/* check if a set of columns determine another column recursively */
Qualify(determinantCols, col, inspected, basicFDs)
    /* reflexivity axiom */
    if col in determinantCols
         return true
    /* this column has already been inspected */
    if col in inspected
         return inspected[col]
    inspected[col] ← false
  /* retrieve all the determinant sets of col. it is possible that one
     column can be determined by multiple determinant sets */
   dtSets ← GetAllDeterminants(col, basicFDs)
    foreach set in dtSets
       qualify ← true
        /* if all the columns within this set can be determined,
         then col can be determined according to transitivity axiom */
       foreach newCol in set
            if Qualify(determinantCols, newCol, inspected, basicFDs) is false
                 then qualify ← false
                      break
       if qualify is true
            then inspected[col] ← true
                 return true
   return false
```

Table 2 depicts the pseudo code of an efficient algorithm (IsDependent) of FD checking.

Lemma 2. Time complexity of IsDependent is $O(lD) \ll O(D^2)$, here *D* is the number of columns, and $l = \max_i |X_i|$, where $|X_i|$ refers to the cardinality of a set X_i.

Proof: similar to the analysis of graph traverse, we use a Boolean array *inspected* to record which column has been evaluated. Thus, each column will be evaluated at most once. And considering we inspect at most D columns, then we come up with the complexity $O(lD)$. Further considering in general, the FDs obtained from data schema describes FD relationship between a small set of dimensions, thus $l \ll d$, the proof concludes.

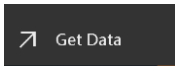
QuickInsights on Complex Datasets in Power BI

Power BI engineering team has supported the query API which QuickInsights needs, so that QuickInsights is able to run against general datasets. In this website, Power BI provides eight sample

datasets for users to get to know how to play with Power BI. Here we also take one dataset “Human Resources Sample” as an example, to show how users can generate insights from this dataset.

Generate insights of “Human Resources Sample”

Step1: go to website https://powerbi.microsoft.com and sign in (or “Sign up free” if you don’t have an account).

Step2: click on the 

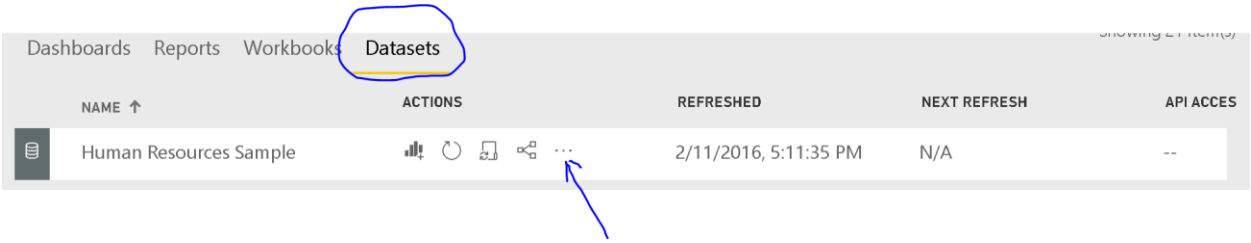
at the bottom left of the web page.

Step3: scroll down and choose Samples and click.

Step4: click on the “Human Resources Sample” and click “Connect”.

It then takes a couple of seconds saying “importing data”

Step 5: switch to “Datasets”, and find “Human Resources Sample”, and click on the “…”



Step6: in the drop-down list, click on “Get quick insights”

After 5~20 seconds, you will see the recommended insights for you!

Top 9 insights for “Human Resources Sample”

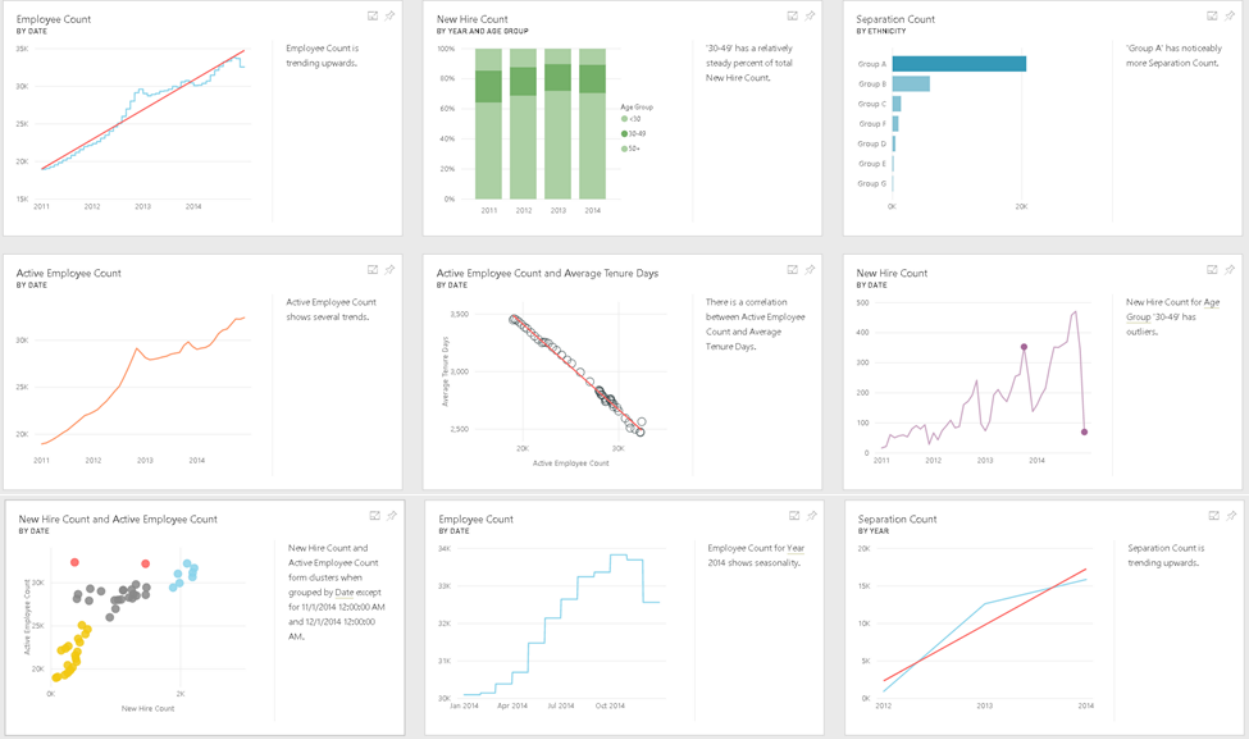


Figure 1. top 9 insights you will see from "Human Resources Sample"

Data Schema of “Human Resources Sample”

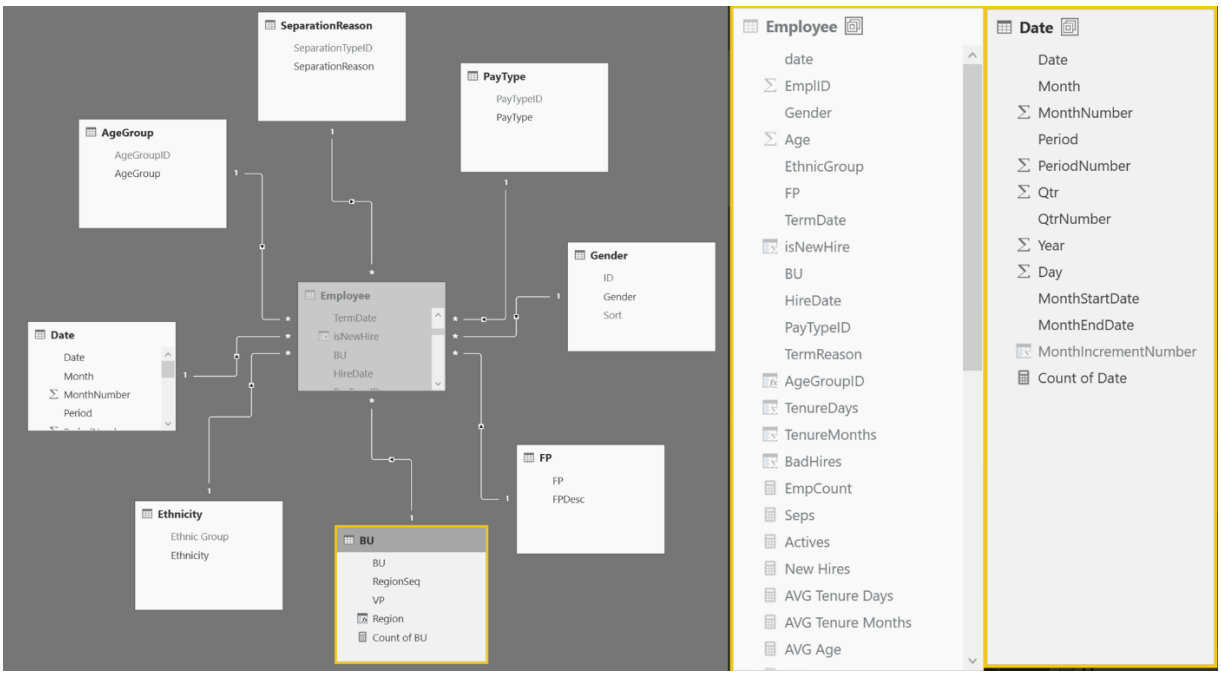


Figure 2. Schema of underlying tables of "Human Resources Sample"

Figure2 shows the database schema of “Human Resources Sample”, you can see it is a typical real-world dataset that consists of several tables, records the information related to employees with various attributes. Power BI engineering team supports the query API that QuickInsights needs, so QuickInsights no longer needs to take care of these underlying complexities. For QuickInsights, it looks like as if

QuickInsights is mining over a large multi-dimensional dataset (joined over all these tables). The implementation of the query API is super efficient (e.g., batched query, cache, and leverage the advantages of Analysis Service) so that QuickInsights in general could search a large portion over the search space.

If you have interests, you can play with all these sample datasets, or even trying your own!

References

[1] Jiawei Han, Jian Pei, Guozhu Dong, and Ke Wang. Efficient Computation of Iceberg Cubes with Complex Measures. SIGMOD, 1-12, 2001.

[2] H. Yao, H. J. Hamilton. Mining functional dependencies from data. DMKD, 197-219, 2008.

[3] M. Y. Vardi. Fundamentals of dependency theory. In E. Borger, editor, Trends in Theoretical Computer Science, 171-224, 1987