A Method for Automated Pairwise Relationship Analysis

Justin Skycak
4 August 2016

1. Introduction and Overview

Automation is a key component of data science - currently, an efficient data science workflow involves programming computers to gather, clean, and unify data so that human intelligence can focus on generating hypotheses, testing hypotheses, and communicating insights to clients. Even in these tasks of human intelligence, tools are used to automate as many sub-tasks as possible - testing hypotheses is supported by R and Python's statistical packages, and communicating insights is backed by Tableau's visualization interface.

However, the process of hypothesis generation is still by-and-large manual. Most hypotheses are generated by clients and domain experts who know what to look for, or analysts who - indirectly, and often by chance - discover an unexpected relationship between variables during their analysis.

With an eye toward automating the process of hypothesis generation, this report introduces a method for exploring pairwise relationships in columnar datasets. The method involves computing relationship metrics for each pair of variable categories in the dataset, and storing these metrics in a new dataset which analysts can filter to pick out relationships of interest.

On the whole, this pairwise relationship analysis method would enable analysts to find useful relationships between variable categories by simply filtering a table of relationship metrics, rather than manually guessing-and-checking numerous pairs. Consequently, it is hoped that this method finds potential use as a future feature in Aunsight.

2. Notation

- Single capital letters (e.g. X, Y) refer to *variables*, while single lower letters (e.g. x, y) refer to *instances*.
- The number of observations with $X = x$ is called the *count* of X and is denoted $n(X = x)$. When unambiguous, we will drop the $X =$ and write only $n(x)$.

- The ratio of $n(x)$ to the total number N of observations is the *probability* of x and is denoted $P(x)$. Explicitly, $P(x) = n(x)/N$.
- The conditional probability of $X = x$ given that $Y = y$ is denoted $P(X = x|Y = y)$, or $P(x|y)$ where unambiguous. Conditional probabilities are computed by Bayes' rule, $P(x|y) = P(x, y)/P(y)$.

3. Pairwise Relationship Analysis

3.1 Motivating Example

Suppose that we attend a 100,000-person Lord of the Rings convention, and we want to know whether there is a relationship between being named Gandalf and liking the color gray at this convention (in Lord of the Rings, there is an esteemed character named Gandalf the Gray). If being named Gandalf and liking gray were unrelated occurrences, we would expect the number of gray-liking Gandalfs to be the product of 100,000, the probability of being named Gandalf, and the probability of one's favorite color being gray. For example, if $P(\text{Gandalf}) = 0.01$ and $P(\text{gray}) = 0.1$, then we would expect there to be $100,000 * 0.01 * 0.1 = 100$ gray-liking Gandalfs. (Note that all probabilities would have to be in reference to this particular gathering, not the nation or world in general.)

If we observed 100 gray-liking Gandalfs at the gathering, we would have little reason to believe that being named Gandalf and liking the color gray had anything to do with one another. However, if we observed 1000 gray-liking Gandalfs, then we would have reason to question whether the two variables were related, since we would have observed $\frac{1000 - 100}{100} = 9$ times more gray-liking Gandalfs than expected.

Suppose that we indeed observed 1000 gray-liking Gandalfs and suspected a relationship between being named Gandalf and liking gray. Our next question might be about the *direction* of the relationship - how often are gray-likers named Gandalf, and how often do Gandalfs like gray? These questions are answered by the conditional probabilities $P(\text{Gandalf} \mid \text{gray})$ and $P(\text{gray} \mid \text{Gandalf})$.

We could ask the same questions even if we had little evidence for a relationship between being named Gandalf and liking gray. However, in this case, a high value for $P(\text{gray} \mid \text{Gandalf})$ would indicate only that many people at the gathering liked gray, i.e. that

$P(\text{gray})$ was high. Although this information might be useful for other analyses, it would not describe any relationship between being named Gandalf and liking gray.

Looking back, the major components of our analysis were

1. *Relationship Existence*: Is there any relationship between being named Gandalf and liking gray? By comparing the observed count of gray-liking Gandalfs to the count we would expect if there was no relationship, we might realize that we observed 9 times more gray-liking Gandalfs than expected, and consequently ought to suspect a relationship.

2. *Relationship Direction*: If someone likes gray, how sure can we be that their name is Gandalf, and if someone is named Gandalf, how sure can we be that they like gray? These questions are answered by the conditional probabilities $P(\text{Gandalf} \mid \text{gray})$ and $P(\text{Gandalf} \mid \text{gray})$.

By automating the process of calculating and displaying the quantities referenced above for every combination of name and color, one could instantly generate hypotheses about all pairwise name-color relationships at once. Such a workflow would be vastly more efficient than relying on intuition and manual labor to decide which name-color pairs to explore and how to explore them.

3.2 Formal Treatment

The computations involved in calculating conditional probabilities are given by Bayes' rule in Sec. 2; therefore, we will skip any formal treatment of relationship direction and focus on a formal treatment of relationship existence.

In general, to determine whether a relationship exists between category x and category y, we compare the observed count $n(x, y)$ to the count $\hat{n}(x, y) = N * P(x) * P(y)$ we would expect if x and y were unrelated. We call this comparison the *relative discrepancy*, and define it by

$$D(x, y) = \frac{n(x, y) - \hat{n}(x, y)}{\hat{n}(x, y)}.$$

The relative discrepancy can be interpreted as follows:

- $D(x, y) = 0.5$ means that x and y occur together 50% more often than expected, so they have a positive association.

- $D(x, y) = 0$ means that x and y occur together as often as we'd expect, so they have no association.
- $D(x, y) = -0.5$ means that x and y occur together 50% less often than we'd expect, so they have a negative association.
- $D(x, y) = -1$ means that x and y do not occur together, so they may be incompatible.

The magnitude of the relative discrepancy measures the strength of a relationship between the two categories, and the sign tells whether the association is positive or negative.

Lastly, the relative discrepancy is ubiquitous. By rearranging the relative discrepancy equation one can see that the relative discrepancy also represents

$$\frac{P(x|y) - P(x)}{P(x)},$$

The relative increase in probability of x associated with y (or y with x), and

$$\frac{P(x|y) - \hat{P}(x|y)}{\hat{P}(x|y)},$$

the relative discrepancy in conditional probability of x given y (or y given x). The relative discrepancy also appears in many standard statistical quantities, such as chi-squared

$$\chi^2 = \frac{[n(x, y) - \hat{n}(x, y)]^2}{\hat{n}(x, y)} = D(x, y)^2 * \hat{n}(x, y)$$

and mutual information

$$I(x, y) = \log \frac{P(x, y)}{P(x)P(y)} = \log[D(x, y) + 1],$$

and could pave the way to the use of more advanced statistics and measures in the future.

3.3 Worked Example

Here, we will demonstrate the calculation of the relative discrepancies and conditional probabilities for a three-column dataset of ten observations:

X	Y	Z
a	*g*	*w*
b	*f*	*w*
b	*f*	*w*
b	*f*	*y*
a	*g*	*y*
a	*g*	*w*
b	*g*	*y*
a	*g*	*y*
a	*g*	*y*
b	*f*	*w*

First, we record the observed count for each pair:

		Y	
		f	*g*
X	*a*	0	5
	b	4	1

		Z	
		w	*y*
Y	*f*	3	1
	g	2	4

		Z	
		w	*y*
X	*a*	2	3
	b	3	2

Next, we calculate the expected count. An easy way to carry out this calculation for a particular row-column intersection is to multiply the sum of observed counts in the row by the sum of observed counts in the column, and then divide by 10.

		Y	
		f	*g*
X	*a*	2	3
	b	2	2.5

		Z	
		w	*y*
Y	*f*	2	2
	g	3	3

		Z	
		w	*y*
X	*a*	2.5	2.5
	b	2.5	2.5

The relative discrepancies are as follows:

		Y	
		f	*g*
X	*a*	-1	0.7
	b	1	-0.6

		Z	
		w	*y*
Y	*f*	0.5	-0.5
	g	-0.3	-0.3

		Z	
		w	*y*
X	*a*	-0.2	0.2
	b	0.2	-0.2

We see that, for example, in the leftmost table,

- *b* and *f* occur together 100% more often than expected,
- *a* and *f* occur together 100% less often than expected (i.e. they never occur together),
- *a* and *g* occur together 70% more often than expected, and
- *b* and *g* occur together 60% less often than expected, and

These relationships are easy to verify by looking back at the original dataset.

Lastly, we calculate observed conditional probabilities for rows given columns (top table) and columns given rows (bottom table). An easy way to calculate the observed conditional probability of a row given a column is to divide the observed count of the pair by the sum of observed counts in the column; similarly, an easy way to calculate the conditional probability of a column variable given a row is to divide the observed count of the pair by the sum of observed counts in the row. Calculations have been rounded to the nearest tenth for simplicity.

		Y	
		f	*g*
X	*a*	0	0.8
	b	1	0.2

		Z	
		w	*y*
Y	*f*	0.6	0.2
	g	0.4	0.8

		Z	
		w	*y*
X	*a*	0.4	0.6
	b	0.6	0.4

		Y	
		f	*g*
X	*a*	0	1
	b	0.8	0.2

		Z	
		w	*y*
Y	*f*	0.8	0.3
	g	0.3	0.7

		Z	
		w	*y*
X	*a*	0.4	0.6
	b	0.6	0.4

We see that, for example, in the leftmost tables,

- *b* occurs 100% of the time that *f* does, and *f* occurs 80% of the time that *b* does.
- Neither *a* nor *f* occurs with the other.
- *a* occurs 80% of the time that *g* does, and *g* occurs 100% of the time that *a* does.
- Both *b* and *g* occur with the other 20% of the time.

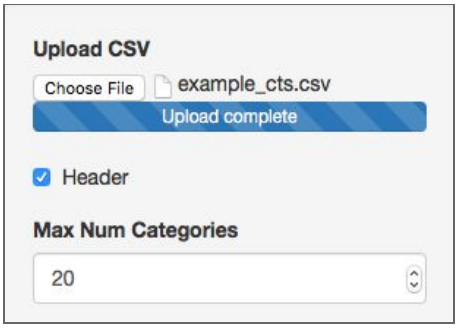
Again, these relationships are easy to verify by looking back at the original dataset.

4. Initial Prototype

Using the R package Shiny, I prototyped an in-browser interface to perform the following steps:

1. Receive data and remove high-dimensional variables
Accept a columnar CSV dataset from the user, and remove high-dimensional variables (such as unique IDs) according to a user-defined cutoff on number of categories. High-dimensional variables would greatly increase the number of category pairs to be analyzed, thus blowing up runtime. Furthermore, relationships between high-dimensional sparse variables are significantly harder to leverage than relationships between low-dimensional dense variables.

letter	fruit	number	name	a	b	fuzzy_a	fuzzy_asquare
a	apple	4	Gandalf	23.4988938	0.270906626	23.5212252	5.121795e+02
a	pear	3	Dave	-5.2655212	0.603606302	-5.6663312	2.737767e+01
b	peach	3	Justin	5.0006853	-0.512832409	4.7891862	1.927668e+01
a	banana	1	Alex	15.4829809	-0.359945009	15.2394958	2.604390e+02
c	apple	1	Justin	11.2502097	0.058495348	11.3690147	1.284028e+02
c	peach	4	Dave	11.9254807	-1.765647511	12.7180646	1.534374e+02
b	peach	4	Dave	21.3740610	-0.425210046	21.5072020	4.390136e+02
a	banana	1	Brian	21.3911908	-0.599641070	20.5487060	4.592984e+02
a	peach	2	Brian	6.7978439	1.162845973	6.7630820	5.629727e+01
c	banana	4	Gandalf	16.6441051	-2.331963706	15.9863765	3.043162e+02
a	apple	4	Gandalf	13.2624568	-3.084996119	13.5912808	1.705418e+02
a	pear	3	Dave	12.8397699	0.401308081	12.2629675	1.664235e+02
b	peach	3	Justin	11.0648861	-0.013121161	11.3454953	1.188863e+02



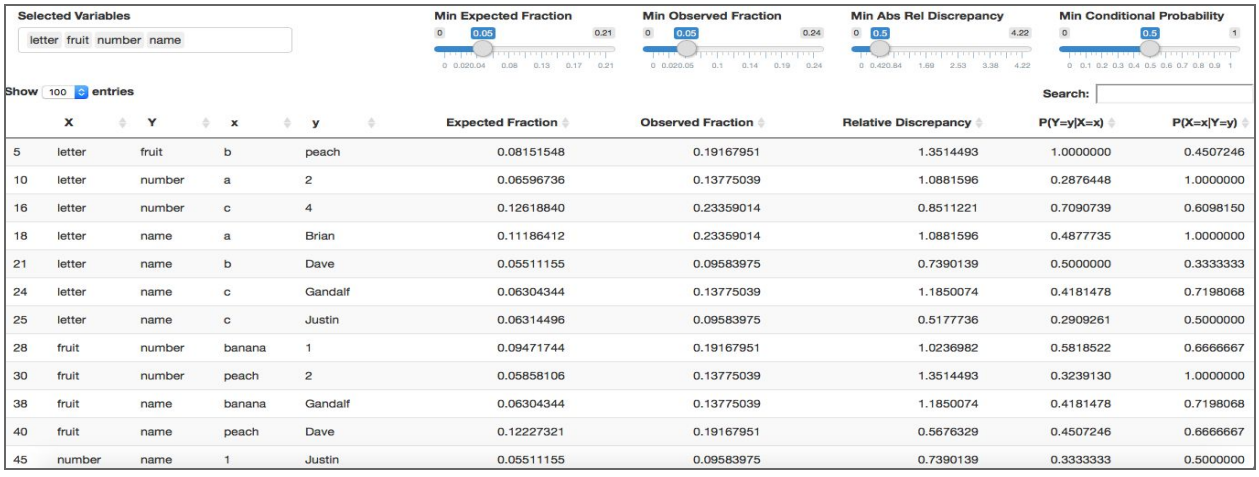
The left image above shows an example input dataset, `example_cts.csv`, consisting of 3245 observations of 8 variables: `letter`, `fruit`, `number`, `name`, `a`, `b`, `fuzzy_a`, and `fuzzy_asquare`. The right image shows the uploading interface, with the maximum number of categories set at 20. The variables `letter`, `fruit`, `number`, and `name` all have fewer than 20 categories; therefore, they are kept. The variables `a`, `b`, `fuzzy_a`, and `fuzzy_asquare` all have more than 20 categories (they are continuous random numbers); therefore, they are removed.

2. Record pairwise relationships

For each pair of selected variables, for each pair of categories, compute the relative discrepancy and conditional probabilities. Store these values in a new pairwise relationship dataset whose columns correspond to variable X, variable Y, category x, category y, observed count, expected count, relative discrepancy, P(x|y), and P(y|x).

3. Display pairwise relationships

Display the pairwise relationship dataset as tabular output, which can be dynamically filtered by any of its columns using tag boxes (for variable and category names) or sliders (for discrepancy fraction and conditional probabilities). Also allow the user to download a copy of the full pairwise relationship dataset.



	X	Y	x	y	Expected Fraction	Observed Fraction	Relative Discrepancy	P(Y=y|X=x)	P(X=x|Y=y)
5	letter	fruit	b	peach	0.08151548	0.19167951	1.3514493	1.0000000	0.4507246
10	letter	number	a	2	0.06596736	0.13775039	1.0881596	0.2876448	1.0000000
16	letter	number	c	4	0.12618840	0.23359014	0.8511221	0.7090739	0.6098150
18	letter	name	a	Brian	0.11186412	0.23359014	1.0881596	0.4877735	1.0000000
21	letter	name	b	Dave	0.05511155	0.09583975	0.7390139	0.5000000	0.3333333
24	letter	name	c	Gandalf	0.06304344	0.13775039	1.1850074	0.4181478	0.7198068
25	letter	name	c	Justin	0.06314496	0.09583975	0.5177736	0.2909261	0.5000000
28	fruit	number	banana	1	0.09471744	0.19167951	1.0236982	0.5818522	0.6666667
30	fruit	number	peach	2	0.05858106	0.13775039	1.3514493	0.3239130	1.0000000
38	fruit	name	banana	Gandalf	0.06304344	0.13775039	1.1850074	0.4181478	0.7198068
40	fruit	name	peach	Dave	0.12227321	0.19167951	0.5676329	0.4507246	0.6666667
45	number	name	1	Justin	0.05511155	0.09583975	0.7390139	0.3333333	0.5000000

5. Quantizing Continuous Variables

The method outlined in this report considers only categorical variables. However, continuous variables can be quantized into different levels, which may be interpreted as categories. For example, a continuous variable taking on values between 0 and 99 might be quantized into low (0-33), medium (33-67), and high (67-100) quantiles.

By treating the quantiles as unordered categories, any information about the ordering of the quantiles is lost. Therefore, unless rank-based methods are incorporated into this method of analysis, it is essential to keep the number of quantiles low. Three quantiles - corresponding to low, medium, and high - seems intuitive and reasonable for most variables, and insights of the form “high X and high Y occur together less often than expected because X tends to be low whenever Y is high” can be refined through additional problem-specific analysis.

Quantizing can be accomplished through k-means clustering algorithms, such as the R package Ckmeans.1d.dp, which is optimized for one-dimensional data. However, it may be challenging to decide whether a given numeric column consists of ordinal continuous data or non-ordinal numeric labels without additional user input.