Correlations & Hypothesis Testing
(with an Appendix on Data Display Tips)
Sheila Kannappan, June 9, 2015

Copy all ".py" files in https://github.com/capprogram/CorrHypTestTutorial to your own working space – these files include partial answers to the exercises below, left incomplete for you to finish. Select exercises have solutions provided with a ".solns" extension.

Given data, we can answer two basic statistical questions without any modeling:

- Are two properties dependent on each other?

  can be answered using a correlation test, which is one type of hypothesis test
- Does the distribution of values of a property differ between two samples?

  can be answered using a distribution test, which is another type of hypothesis test

Note that neither question necessarily involves a model (such as a function with parameters) – these are basic yes/no questions, in practice answered with a probability.

I. Correlation Tests

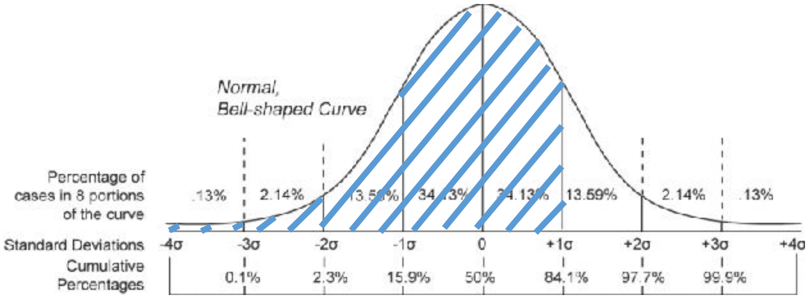
The two most common correlation tests are the Pearson Correlation test, which assumes a linear relation with Gaussian scatter, so actually *is* parametric, and the Spearman Rank test, which assumes only a monotonic relation, and is therefore non-parametric. The "correlation coefficients" returned by these tests measure the strength of the correlation or anti-correlation (yielding a negative coefficient).

(a) Use *plt.subplot* to create plots of "Anscombe's Quartet," four data sets with identical Pearson correlation coefficients, all scattered around $y = 3 + 0.5x$. There are two lessons we can draw from Anscombe's Quartet:

1) *Always look at your data.* How can it be that the Pearson test result is the same in all cases? Think about why this is mathematically possible.

2) *Choose the right test for the job, based on the assumptions built into the test.* How well do each of the four Quartet data sets satisfy the assumptions of the Pearson and Spearman tests? When is one or the other better?

(b) Both the Pearson and Spearman tests can return the *probability of no correlation* p_{null} in addition to/instead of the correlation coefficient. With it, we can interpret the probability that there *is* a correlation, $1- p_{null}$, in terms of familiar confidence levels from a Gaussian distribution by performing the integral of the probability (area) under a normalized ± =1 Gaussian from $-\infty$ up to x, where x is the value that allows the integral to equal $1- p_{null}$. Then if $x=\#$ we say we have a $\#\sigma$ confidence correlation (remember $\sigma=1$). For example if $p_{null}=5\%$ then we have 95% in our detection of a correlation, which we might at first assume is like a "2σ result" remembering that $\pm2\sigma$ contains 95% of the area under a Gaussian. However, there is one subtlety here, in that detection of a positive signal standing out from random noise involves a "one-sided" confidence interval, meaning we integrate up from –∞ to x, rather than a "two-sided" confidence interval, as would be used when expressing a measurement ± an error. So for example, the standard ±1σ error confidence interval is a two-sided, 68% confidence interval, whereas a 68% confidence *detection* does not reach 1σ – as seen in the diagram below, an 84% confidence detection is a 1σ result. Thus technically if $p_{null}=5\%$, our one-sided "detection" confidence level is not 2σ but 1.6σ, which is logically also the upper boundary of a two-sided interval containing 90% of the probability.



Use *stats.norm.interval* to prove this last statement (it computes two-sided intervals by default). Once you understand this function's behavior, use it to convert the % confidence level associated with each correlation to the equivalent σ confidence level, and print this level on each panel for the Pearson and Spearman tests separately.

(c) Finally, analyze the code used to create the plots in this exercise. What helpful data display strategies have been employed? (Consider especially #s 8-13 in the Appendix.)

Advanced topics. Browse the web to learn about these issues that complicate correlation tests:

- selection bias
- covariance
- causality/hidden parameters
- third parameters/partial correlations

Note that multi-parameter data sets are often analyzed using "principal component analysis" (PCA) to find the most fundamental driving parameters. As designed, PCA is most effective for <u>linear</u> correlations. Google "PCA" to learn more.

II. Distribution Tests

Two other common hypothesis tests are the Kolmogorov-Smirnov (K-S) and Chi-Squared (χ^2) tests. Each has multiple versions/definitions, but we will focus on the most basic applications.

The K-S test is useful for determining whether two data distributions were drawn from the same or different parent populations. It returns the probability of the null hypothesis (i.e., that they were drawn from the same parent population), and $1\text{-}p_{null}$ is then the confidence in the result that the two distributions are different.

The χ^2 test involves computing the χ^2 value, which is useful for determining whether a model is consistent with a data set within its errors. Most (astro)physicists define it as $¡^2 = \sum_i \frac{(O_i - E_i)}{\pm_i^2}$ where O_i, E_i, and $\pm_i$ are the Observed and Expected values and the errors. In other words, the numerator represents the actual residuals between the data and the model, and the denominator represents the expected residuals assuming Gaussian-distributed errors. (Note however that in statistics, χ^2 is generally defined with an E_i in the denominator, which represents the special case of Poisson-distributed data. If this distribution is unfamiliar to you, don't worry about it.)

To see how the χ^2 value can serve as a test, consider that if the model is correct and the errors have been correctly estimated, $\chi^2 \approx$ N, where N is the number of degrees of freedom (number of data points minus number of parameters in the model). Therefore scientists often speak loosely and say that if the "reduced" Chi-squared defined as χ^2/N is approximately equal to 1, then the fit is good. But let's take a closer look.

(a) Using Monte Carlo methods, create 1000 fake data sets following the underlying functional form *y=1/x* for *x* = 1, 2, 3…30 with Gaussian random errors on *y* of amplitude 0.1. Each data set defines one value of χ^2, and the 1000 values of χ^2 from all of the data sets can be divided by N and binned into a histogram to show you the reduced χ^2 distribution, which is a well-defined function analogous to a Gaussian or any other function. Note that *y=1/x* has no free parameters, so N is just the # of data points, 30.

(b) Now create 1000 fake data sets each with 300 values of *x* = 1.1, 1.2, 1.3… 30.9, using the same function *y=1/x* with the same errors of 0.1 on *y*. Overplot the new histogram of reduced χ^2 values for N=300. Is a reduced χ^2 of 1.3 equally good for both data sets?

(c) Use a K-S test (stats.ks_2samp) to quantify your confidence level that the two χ^2 distributions are different. Google the functional form of the χ^2 distribution on the web to understand why in mathematical terms.

(d) This exercise shows that just knowing that the reduced $\chi^2 \approx 1$ does not tell you how good your model is. You must know N. If you do, you can compute confidence levels by integrating the probability under the normalized χ^2 distribution up to your measured χ^2. Use np.argsort to do this approximately with the χ^2 distributions from your Monte Carlo.

(e) Again, analyze the code used to create the plots in this exercise. What helpful data display strategies have been employed?

Advanced Topics. When comparing models, you can use the fact that the probability or "likelihood" $\mathscr{L}$ that a given model is correct is proportional to $e^{-¡^2/2}$, and the likelihood ratio $\mathscr{L}_1/\mathscr{L}_2$ formed with two different χ^2 values from two different models describes our confidence in one model relative to the other. Such comparisons form the basis of Bayesian statistics. Google "likelihood ratio test" or "Bayesian statistics" to learn more.

Appendix: Data Display Tips for Talks & Posters (or General Purpose: #8-13)

1. If making a poster, set the custom paper size in powerpoint/keynote before you start.
2. A typical poster format starts with a "big picture" Intro in the top left panel, mixing general background and an abstract, and ends with a Conclusions & Future Work section in the bottom right panel. Acknowledgements also fit at the bottom (e.g. funding sources). Coauthors are listed under the title of a poster but may be listed at any appropriate time for talks. Give only essential references, kept brief and/or small.
3. Talks are most readable with light text on a dark background, whereas posters are the opposite. Stark white is hard on people's eyes in all cases, so a light gray or pale yellow is preferred. Try to harmonize/simplify the overall palette. Also remember many people are red-green colorblind.
4. **Minimize text**, have large rather than many figures, and maximize white space. Use phrases and bullets rather than sentences whenever natural.
5. Section headings and figures alone should tell most of the story. Likewise, reading the Intro and Conclusions/Future Work should be sufficient to give the highlights.
6. Seraphless fonts like this are most readable from a distance. Fonts should be large enough:
    a. posters: 20pt for references, 32pt for normal text, 48pt for section headings, 92pt for banner title
    b. talks: nothing smaller than 20pt except references
7. Use changes in font type/italics/boldface judiciously, e.g. to distinguish captions from the main text. Likewise use color in judicious and intuitive ways – excess color and font diversity is distracting.
8. Decide what the main message of each figure is, and consider making duplicates with different information highlighted if there are several messages.
9. If the best way to communicate the message isn't obvious, sketch/discuss the content with collaborators, and experiment (contours vs. shading, histograms vs. plots, etc.).
10. Within figures, double up information (e.g. shape + color, symbol size + graying, histogram color + fill type, etc.). The result should leap out without much thought.
11. Add legends, key info, labeling to clarify meaning of lines/contours/fits, etc, so that ideally the caption becomes almost unnecessary….
12. …But at the same time, avoid clutter. Error bars can be placed in a corner if one representative example is good enough.
13. Follow accepted practice in your field (which axis is x or y, tick direction, etc.…)
14. Ideally start figure captions with a short sentence fragment expressing overall content.
15. When taking figures from articles, overlay readable axes with large fonts. Simplify axis labeling with cartoons or "plain English." Thicken lines (axes, fits, fonts, etc.).
16. Discuss drafts with mentors, group members, roommates. Leave time for changes.