BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN

TIỂU LUẬN
CƠ SỞ TOÁN CHO HỌC MÁY

Tên tiểu luận: Chương 7 - Mối quan hệ giữa các biến

Giảng viên HD: **TS. Nguyễn Văn Hậu**

Học viên thực hiện: **Nông Quang Huy**

Lớp: **H01222**

Hưng Yên, 6/2023

LỜI CẢM ƠN

Trong suốt quá trình học tập, và hoàn thành tiểu luận môn **Cơ sở toán cho học máy**, chúng em đã nhận được rất nhiều sự hướng dẫn tận tình quý báu của thầy cô. Với lòng biết ơn sâu sắc em xin được bày tỏ lời cảm ơn đến:

- Ban giám hiệu Trường Đại học sư phạm kỹ thuật Hưng Yên, khoa Công nghệ thông tin đã tận tình giúp đỡ và tạo điều kiện cho em trong quá trình học tập.
- Đặc biệt, chúng em xin cảm ơn thầy **Nguyễn Văn Hậu -** người đã trực tiếp giảng dạy, hướng dẫn và tạo mọi điều kiện để em thực hiện bài tiểu luận này bằng tất cả lòng nhiệt tình và sự quan tâm sâu sắc.
- Xin chân thành cảm ơn các thầy cô trong khoa Công nghệ thông tin đã tạo cơ hội cho em được học tập, nghiên cứu và tích lũy kiến thức để thực hiện bài tiểu luận.

Em đã cố gắng vận dụng những kiến thức đã học được và tìm tòi thêm nhiều thông tin để hoàn thành bài tiểu luận này. Tuy nhiên, do kiến thức còn hạn chế và chưa có nhiều kinh nghiệm trên thực tiễn nên khó tránh khỏi những thiếu sót trong bài làm. Rất kinh mong quý thầy, cô cho em thêm những góp ý để bài tiểu luận của em được hoàn thiện hơn.

Chúng em xin chân thành cảm ơn!

MỤC LỤC

CHƯƠNG 7: MỐI QUAN HỆ GIỮA CÁC BIẾN

Cho đến nay, chúng ta chỉ xem xét một biến tại một thời điểm. Trong chương này, chúng ta sẽ tìm hiểu về các mối quan hệ giữa các biến. Hai biến có mối quan hệ nếu việc biết một biến cung cấp thông tin về biến kia. Ví dụ, chiều cao và cân nặng có liên quan với nhau; những người cao hơn có xu hướng nặng hơn. Tất nhiên, đó không phải là một mối quan hệ hoàn hảo: có những người thấp nhưng nặng và những người cao nhưng nhẹ. Nhưng nếu bạn muốn đoán khối lượng của ai đó, bạn sẽ chính xác hơn nếu bạn biết chiều cao của họ hơn là không biết. Đoạn mã cho chương này được đặt trong file scatter.py. Để biết thông tin về tải xuống và làm việc với mã này, xem phần “Using the Code” ở trang xi.

Biểu đồ phân tán

Cách đơn giản nhất để kiểm tra mối quan hệ giữa hai biến là biểu đồ phân tán, nhưng tạo ra một biểu đồ phân tán tốt không phải lúc nào cũng dễ dàng. Ví dụ, tôi sẽ sẽ vẽ biểu đồ phân tán của cân nặng so với chiều cao cho các người trả lời trong BRFSS (xem "The lognormal Distribution" trang 55).

Đây là đoạn mã đọc file dữ liệu và trích xuất chiều cao và cân nặng:

```
df = brfss.ReadBrfss(nrows=None)
sample = thinkstats2.SampleRows(df, 5000)
heights, weights = sample.htm3, sample.wtkg2
```

SampleRows chọn một tập hợp con ngẫu nhiên của dữ liệu:

```
def SampleRows(df, nrows, replace=False):
indices = np.random.choice(df.index, nrows, replace=replace)
sample = df.loc[indices]
return sample
```

df là DataFrame, nrows là số hàng để chọn, và replace là một giá trị boolean cho biết liệu việc lấy mẫu nên được thực hiện với sự thay thế; nghĩa là liệu có thể chọn cùng một hàng nhiều lần hay không.

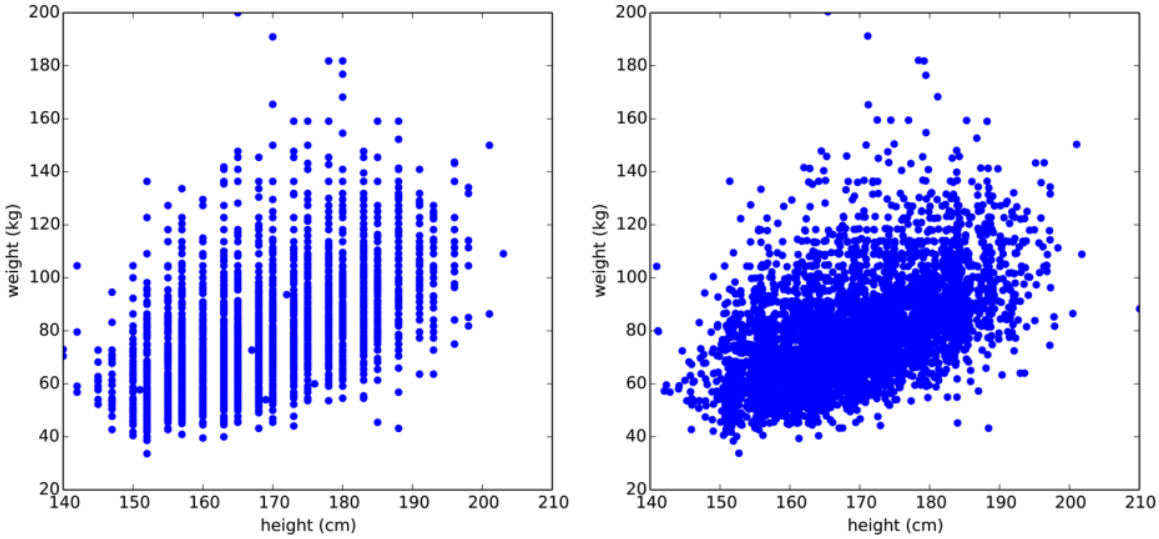
thinkplot cung cấp Scatter, giúp tạo ra các biểu đồ phân tán:

```
thinkplot.Scatter(heights, weights)
thinkplot.Show(xlabel='Height (cm)',
```

ylabel='Weight (kg)',

axis=[140, 210, 20, 200])

Kết quả, trong Hình 7-1 (bên trái), thể hiện hình dạng của mối quan hệ. Như chúng ta mong đợi, những người cao thường có cân nặng lớn hơn.



Hình 7-1. Biểu đồ phân tán về cân nặng so với chiều cao đối của những người trả lời trong BRFSS, không jittered (trái), jittered (phải).

Nhưng đây không phải là phương pháp biểu diễn tốt nhất của dữ liệu, vì dữ liệu được đóng gói vào các cột. Vấn đề là chiều cao được làm tròn đến inch gần nhất, chuyển đổi sang cm và sau đó được làm tròn lại. Một số thông tin bị mất trong quá trình dịch.

Chúng ta không thể lấy lại thông tin đó, nhưng chúng ta có thể giảm thiểu tác động lên biểu đồ phân tán bằng cách jittered dữ liệu, nghĩa là thêm nhiễu ngẫu nhiên để đảo ngược tác động của làm tròn. Khi các phép đo này được làm tròn đến inch gần nhất chúng có thể sai lệch lên đến 0,5 inch hoặc 1,3 cm. Tương tự, cân nặng có thể sai lệch lên đến 0,5 kg.

heights = thinkstats2.Jitter(heights, 1.3)

weights = thinkstats2.Jitter(weights, 0.5)

Đây là cách khai báo Jitter:

def Jitter(values, jitter=0.5):

n = len(values)

return np.random.uniform(-jitter, +jitter, n) + values

Các giá trị có thể là bất kỳ chuỗi nào; kết quả là một mảng NumPy.

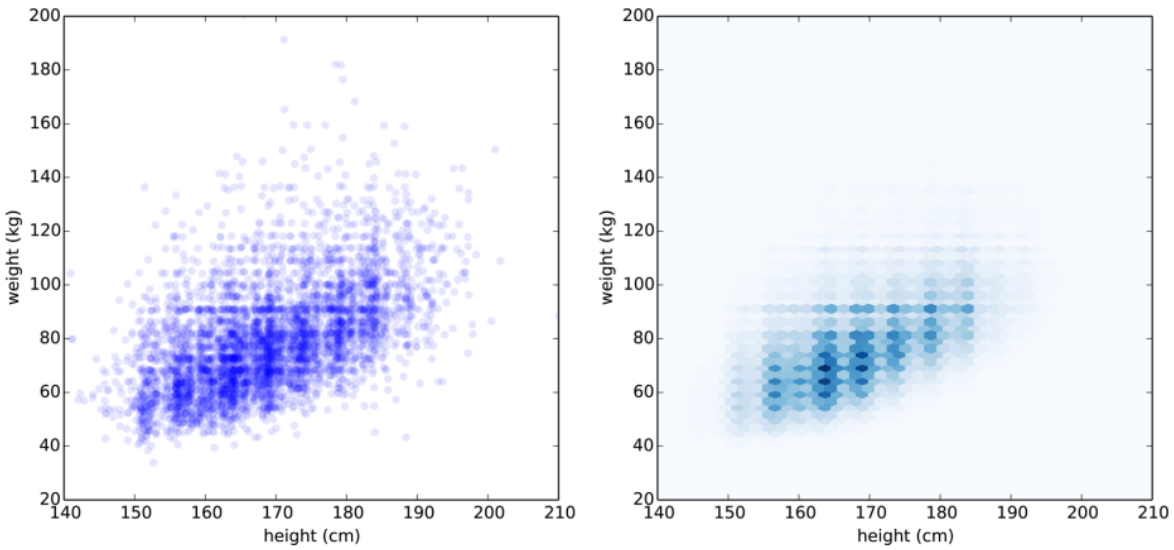
Hình 7-1 (bên phải) cho thấy kết quả. Jittering giảm thiểu tác động trực quan của làm tròn và làm cho hình dạng của mối quan hệ rõ ràng hơn. Nhưng nói chung, bạn chỉ nên jitter dữ liệu cho mục đích trực quan hóa và tránh sử dụng tránh sử dụng jitter dữ liệu cho phân tích.

Ngay cả với Jittering, đây không phải là cách tốt nhất để biểu diễn dữ liệu. Có nhiều điểm chồng lấp nhau, ẩn thông tin trong các phần dày đặc của hình và tạo ra một sự chênh lệch khi nhấn mạnh các ngoại lệ.. Hiệu ứng này được gọi là bão hòa.

Chúng ta có thể giải quyết vấn đề này bằng tham số alpha, điều này làm cho các điểm một phần trong suốt:

```
thinkplot.Scatter(heights, weights, alpha=0.2)
```

Hình 7-2 (trái) cho thấy kết quả. Các điểm dữ liệu chồng lấp tối hơn, vì vậy mức độ tối tỷ lệ thuận với mật độ. Trong đồ thị này, chúng ta có thể thấy hai chi tiết không rõ ràng trước: các nhóm dọc tại nhiều chiều cao khác nhau và một đường ngang gần 90 kg hoặc 200 pounds. Vì dữ liệu này dựa trên các bản tự báo cáo tính bằng pound, nên lời giải thích có khả năng nhất là một số người trả lời các giá trị đã được làm tròn.



Hình 7-2. Biểu đồ phân tán với *jittering* và *transparency* (trái), biểu đồ *hexbin* (phải)

Sử dụng độ trong suốt hoạt động tốt cho các bộ dữ liệu có kích thước trung bình, nhưng hình này chỉ hiển thị 5000 bản ghi đầu tiên trong BRFSS, trong tổng số 414.509 bản ghi. Để xử lý các tập dữ liệu lớn hơn, một lựa chọn khác là biểu đồ hexbin, chia biểu đồ thành ô lục giác và tô màu mỗi ô theo số lượng điểm dữ liệu nằm trong đó. thinkplot cung cấp HexBin:

```
thinkplot.HexBin(heights, weights)
```

Hình 7-2 (bên phải) cho thấy kết quả. Ưu điểm của hexbin là nó hiển thị hình dạng của mối quan hệ tốt và hiệu quả đối với các tập dữ liệu lớn, cả về thời gian và kích thước của tập tin mà nó tạo ra. Một nhược điểm là nó làm cho những giá trị ngoại lệ trở nên không rõ ràng.

Mục đích của ví dụ này là không dễ để tạo một biểu đồ phân tán thể hiện mối quan hệ quan hệ rõ ràng mà không đưa vào các giá trị hiện tượng sai lệch.

Đặc trưng mối quan hệ

Biểu đồ phân tán cung cấp một ấn tượng chung về mối quan hệ giữa các biến, nhưng có nhiều biểu đồ khác cung cấp cái nhìn sâu sắc hơn về bản chất của mối quan hệ. Một trong những lựa chọn là phân loại một biến và vẽ đồ thị phân vị của biến còn lại.

NumPy và pandas cung cấp các hàm để phân loại dữ liệu:

```
df = df.dropna(subset=['htm3', 'wtkg2'])
bins = np.arange(135, 210, 5)
indices = np.digitize(df.htm3, bins)
groups = df.groupby(indices)
```

dropna loại bỏ hàng có giá trị nan trong bất kỳ cột nào trong danh sách được liệt kê. Arange tạo ra một mảng NumPy các bin từ 135 đến, nhưng không bao gồm, 210, với bước nhảy là 5.

digitize tính toán chỉ mục của bin chứa từng giá trị trong df.htm3. Kết quả là một mảng NumPy gồm các số nguyên. Các giá trị nằm dưới bin thấp nhất được ánh xạ tới chỉ mục 0. Các giá trị cao hơn bin cao nhất được ánh xạ đến len(bins)

groupby là một phương thức DataFrame trả về một đối tượng GroupBy; được sử dụng trong một vòng lặp for, groups duyệt qua các tên của các nhóm và

DataFrames đại diện cho chúng. Vì thế, ví dụ: chúng ta có thể in số lượng hàng trong mỗi nhóm như sau:

```
for i, group in groups:
print(i, len(group))
```

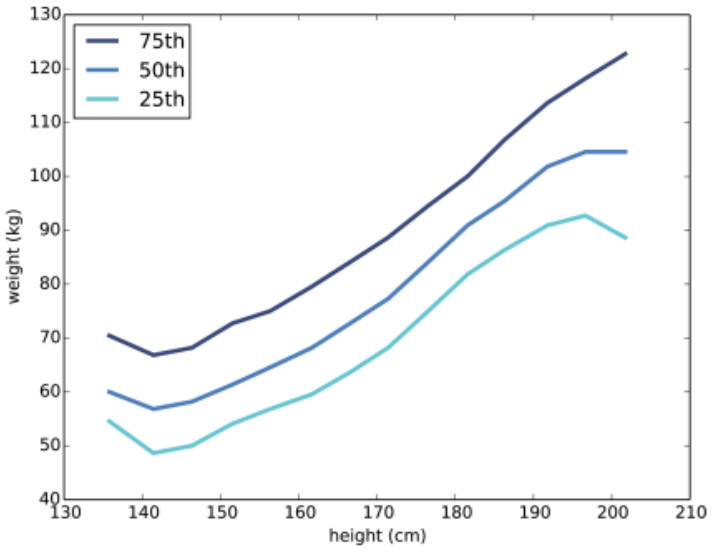
Bây giờ đối với mỗi nhóm, chúng ta có thể tính chiều cao trung bình và CDF của cân nặng:

```
heights = [group.htm3.mean() for i, group in groups]
cdfs = [thinkstats2.Cdf(group.wtkg2) for i, group in groups]
```

Cuối cùng, chúng ta có thể vẽ đồ thị phân vị của cân nặng so với chiều cao:

```
for percent in [75, 50, 25]:
weights = [cdf.Percentile(percent) for cdf in cdfs]
label = '%dth' % percent
thinkplot.Plot(heights, weights, label=label)
```

Hình 7-3 cho thấy kết quả. Trong khoảng từ 140 và 200 cm, mối quan hệ giữa các biến gần như tuyến tính. Phạm vi này bao gồm hơn 99% dữ liệu, vì vậy chúng ta không phải lo lắng quá nhiều về các giá trị cực đoan.



Hình 7-3. Các phân vị của cân nặng cho một loạt các bin chiều cao.

Tương quan

Tương quan là một thống kê nhằm xác định mức độ mạnh yếu của mối liên hệ giữa hai biến

Một thách thức trong việc đo lường mối tương quan là các biến chúng ta muốn so sánh thường không được biểu thị bằng cùng một đơn vị. Và ngay cả khi chúng cùng một đơn vị, chúng có phân phối khác nhau.

Có hai giải pháp thông thường cho các vấn đề này:

• Biến đổi từng giá trị thành điểm chuẩn, là số độ lệch chuẩn từ giá trị trung bình. Sự biến đổi này dẫn đến "hệ số tương quan Pearson theo phép tích moment sản phẩm".

• Biến đổi từng giá trị thành thứ hạng của nó, là chỉ số của nó trong danh sách các giá trị được sắp xếp. Cái này biến đổi dẫn đến "hệ số tương quan Spearman theo thứ hạng."

Nếu X là một chuỗi n giá trị, xi, chúng ta có thể chuyển đổi thành điểm chuẩn bằng cách trừ đi giá trị trung bình và chia cho độ lệch chuẩn: $z_i = (x_i - \mu)/\sigma$. Tử số là độ lệch: khoảng cách so với giá trị trung bình. Chia cho σ chuẩn hóa sai lệch, vì vậy các giá trị của Z là không thứ nguyên (không có đơn vị) và phân phối của nó có trung bình là 0 và phương sai là 1.

Nếu X có phân phối chuẩn thì Z cũng vậy. Nhưng nếu X bị lệch hoặc có các giá trị ngoại lệ thì Z cũng vậy; trong những trường hợp đó, s sử dụng xếp hạng phần trăm sẽ hiệu quả hơn. Nếu chúng ta tính toán một biến mới, R, sao cho r_i là hạng của xi, phân phối của R đều từ 1 đến n, bất kể phân phối của X.

Hiệp phương sai

Hiệp phương sai là thước đo xu hướng của hai biến thay đổi cùng nhau. Nếu chúng ta có hai chuỗi, X và Y, độ lệch của chúng so với giá trị trung bình là:

$$d\,x_i = x_i - \bar{x}$$

$$d\,y_i = y_i - \bar{y}$$

trong đó $\bar{x}$ là giá trị trung bình mẫu của X và $\bar{y}$ là giá trị trung bình mẫu của Y. Nếu X và Y thay đổi cùng nhau, độ lệch của chúng có xu hướng có cùng một dấu. Nếu chúng ta nhân chúng với nhau, tích sẽ có giá trị dương khi độ lệch có cùng dấu và âm khi chúng trái dấu. Do đó, việc cộng tổng của các tích này cho ta một đo lường của xu hướng thay đổi cùng nhau.

Hiệp phương sai là giá trị trung bình của các tích này:

$$Cov(X , Y) = 1n \sum d\ xi\ d\ yi$$

trong đó n là độ dài của hai chuỗi (chúng phải có cùng độ dài).

Nếu bạn đã nghiên cứu về đại số tuyến tính, bạn có thể nhận ra rằng Cov là tích vô hướng của các độ lệch, chia cho độ dài của chúng. Vì vậy, hiệp phương sai đạt giá trị lớn nhất nếu hai vectơ giống nhau, 0 nếu chúng vuông góc và âm nếu chúng ngược hướng nhau. thinkstats2 sử dụng np.dot để triển khai Cov hiệu quả:

```
def Cov(xs, ys, meanx=None, meany=None):
xs = np.asarray(xs)
ys = np.asarray(ys)
if meanx is None:
meanx = np.mean(xs)
if meany is None:
meany = np.mean(ys)
cov = np.dot(xs-meanx, ys-meany) / len(xs)
return cov
```

Theo mặc định, Cov tính sai số từ các giá trị trung bình của mẫu hoặc bạn có thể cung cấp các giá trị trung bình đã biết. Nếu xs và ys là các chuỗi Python, thì np.asarray sẽ chuyển đổi chúng thành mảng NumPy. Nếu chúng đã là mảng NumPy, thì np.asarray không thực hiện gì cả. Việc triển khai hiệp phương sai này có nghĩa là đơn giản cho mục đích giải thích. NumPy và pandas cũng cung cấp triển khai hiệp phương sai, nhưng cả hai đều áp dụng một sự điều chỉnh cho các cỡ mẫu nhỏ mà chúng ta chưa đề cập đến và np.cov trả về một ma trận hiệp phương sai, điều đó nhiều hơn chúng ta cần ở thời điểm hiện tại..

Tương quan Pearson

Hiệp phương sai rất hữu ích trong một số tính toán, nhưng nó hiếm khi được báo cáo dưới dạng tóm tắt thống kê vì nó khó giải thích.

Trong số các vấn đề khác, các đơn vị của nó là sản phẩm của các đơn vị X và Y. Ví dụ: hiệp phương sai của cân nặng và chiều cao trong BRFSS tập dữ liệu là 113 kilôgam-cm, không biết điều đó có nghĩa là gì. Một giải pháp cho vấn đề

này là chia độ lệch cho độ lệch chuẩn, mang lại điểm tiêu chuẩn và tính tích của điểm tiêu chuẩn:

$$p_i = \frac{(x_i - \bar{x})}{S_X} \frac{(y_i - \bar{y})}{S_Y}$$

Trong đó SX và SY là độ lệch chuẩn của X và Y. Giá trị trung bình của các sản phẩm này là

$$\rho = \frac{1}{n} \Sigma p_i$$

Hoặc chúng ta có thể viết lại ρ bằng cách rút gọn SX và SY:

$$\rho = \frac{Cov(X, Y)}{S_X S_Y}$$

Giá trị này được gọi là tương quan Pearson theo tên của Karl Pearson, một nhà thống kê ảnh hưởng đến thời kỳ đầu. Nó rất dễ tính toán và dễ giải thích. Bởi vì điểm tiêu chuẩn là thứ nguyên vô hướng, ρ cũng vậy.

Đây là cách triển khai trong thinkstats2:

```
def Corr(xs, ys):
xs = np.asarray(xs)
ys = np.asarray(ys)
meanx, varx = MeanVar(xs)
meany, vary = MeanVar(ys)
corr = Cov(xs, ys, meanx, meany) / math.sqrt(varx * vary)
return corr
```

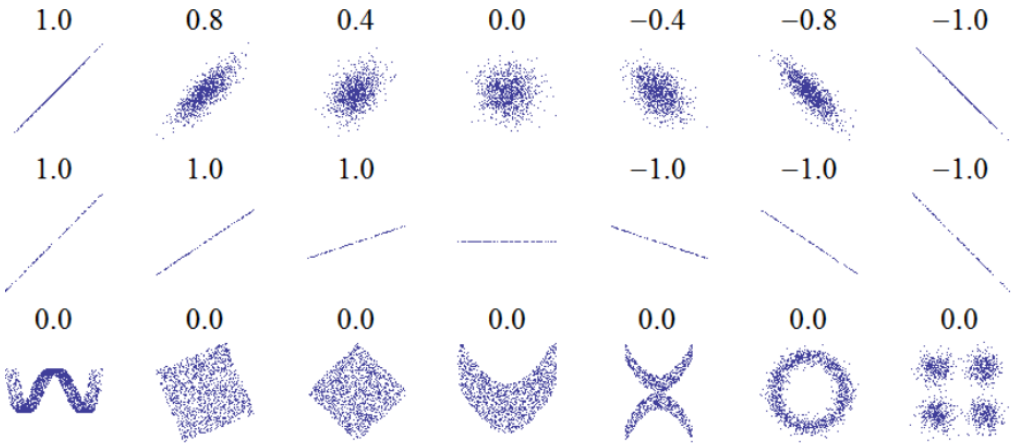
MeanVar tính toán giá trị trung bình và phương sai cho hiệu quả hơn một chút so với việc gọi riêng biệt đến np.mean và np.var. Tương qua Pearson luôn nằm trong khoảng từ -1 đến +1 (bao gồm cả hai). Nếu ρ dương, chúng ta nói rằng tương quan là dương, có nghĩa là khi một biến cao, biến kia có xu hướng cao. Nếu ρ âm, tương quan âm, vì vậy khi một biến cao, biến khác là thấp.

Độ lớn của ρ biểu thị độ mạnh của mối tương quan. Nếu ρ bằng 1 hoặc -1, các biến tương quan hoàn hảo, có nghĩa là nếu bạn biết một biến, bạn có thể dự đoán một cách hoàn hảo về biến khác.

Hầu hết các mối tương quan trong thế giới thực là không hoàn hảo, nhưng nó vẫn hữu ích. tương quan giữa chiều cao và cân nặng là 0,51, đây là mối tương quan chặt chẽ so với các biến tương tự liên quan đến con người.

Mối quan hệ phi tuyến tính

Nếu hệ số tương quan Pearson gần bằng 0, thì có thể kết luận rằng không có mối quan hệ giữa các biến, nhưng kết luận đó không hợp lệ. Hệ số tương quan Pearson chỉ đo lường các mối quan hệ tuyến tính. Nếu có một mối quan hệ phi tuyến tính, ρ sẽ không đánh giá được độ mạnh của nó. Hình 7-4 là từ trang Wikipedia về tương quan và sự phụ thuộc. Nó cho thấy biểu đồ phân tán và hệ số tương quan cho một số tập dữ liệu được xây dựng cẩn thận.



Hình 7-4. Ví dụ về bộ dữ liệu với một loạt các mối tương quan.

Hàng đầu tiên hiển thị các mối quan hệ tuyến tính với một loạt các mối tương quan; bạn có thể sử dụng hàng này để biết các giá trị khác nhau của ρ như thế nào. Hàng thứ hai cho thấy các mối quan hệ hoàn hảo với một loạt các độ dốc, điều này chứng tỏ rằng mối tương quan không liên quan đến độ dốc (chúng ta sẽ sớm nói về việc ước tính độ dốc). Hàng thứ ba cho thấy các biến liên quan rõ ràng với nhau, nhưng vì mối quan hệ này là phi tuyến tính nên hệ số tương quan là 0.

Bài học của câu chuyện này là bạn phải luôn xem xét biểu đồ phân tán dữ liệu của mình trước khi mù quáng tính toán một hệ số tương quan.

Tương quan thứ hạng Spearman

Tương quan Pearson hoạt động tốt nếu mối quan hệ giữa các biến là tuyến tính và nếu các biến gần đúng chuẩn. Nhưng nó không mạnh mẽ khi có sự xuất hiện của các ngoại lệ. Tương quan thứ hạng Spearmanlà một giải pháp thay thế giúp giảm thiểu tác động của các giá trị ngoại lai và phân phối lệch. Để tính tương quan Spearman, chúng ta phải tính thứ hạng của từng giá trị, đó là chỉ số của nó trong mẫu được sắp xếp. Ví dụ, trong mẫu [1, 2, 5, 7] thứ hạng của giá trị 5 là 3, vì nó xuất hiện thứ ba trong danh sách được sắp xếp. Sau đó, chúng tôi tính toán Tương quan của Pearson cho các cấp bậc.

thinkstats2 cung cấp một hàm tính toán tương quan xếp hạng Spearman:

```
def SpearmanCorr(xs, ys):
xranks = pandas.Series(xs).rank()
yranks = pandas.Series(ys).rank()
return Corr(xranks, yranks)
```

Tôi chuyển đổi các đối số thành các đối tượng pandas series để có thể sử dụng thứ hạng, hàm tính toán xếp hạng cho từng giá trị và trả về một Series. Sau đó, tôi sử dụng Corr để tính toán mối tương quan cho các xếp hạng.

Tôi cũng có thể sử dụng trực tiếp Series.corr và chỉ định phương thức Spearman:

```
def SpearmanCorr(xs, ys):
xranks = pandas.Series(xs)
yranks = pandas.Series(ys)
return xs.corr(ys, method='spearman')
```

Tương quan xếp hạng Spearman cho dữ liệu BRFSS là 0,54, cao hơn một chút so với tương quan Pearson, 0,51. Có một số nguyên nhân khả thi cho sự khác biệt này, bao gồm các lý do sau đây:

- Nếu mối quan hệ là phi tuyến tính, tương quan Pearson có xu hướng đánh giá thấp sức mạnh của mối quan hệ.

- Tương quan Pearson có thể bị ảnh hưởng (theo cả hai hướng) nếu một trong các phân phối bị lệch hoặc chứa các ngoại lệ. Mối tương quan xếp hạng của Spearman mạnh hơn.

Trong ví dụ BRFSS, chúng ta biết rằng phân phối trọng lượng gần như bất thường; sau khi áp dụng biến đổi log, nó xấp xỉ một phân phối chuẩn, vì vậy nó không có độ lệch. Vì thế một cách khác để loại bỏ ảnh hưởng của độ lệch là tính toán tương quan Pearson với log-cân nặng và chiều cao:

thinkstats2.Corr(df.htm3, np.log(df.wtkg2)))

Kết quả là 0,53, gần với tương quan thứ hạng, 0,54. Vì vậy, điều đó cho thấy rằng sự sai lệch trong sự phân bổ trọng lượng giải thích phần lớn sự khác biệt giữa tương quan Pearson và Spearman.

Tương quan và nhân quả

Nếu các biến A và B tương quan với nhau, có ba cách giải thích: A gây ra B, hoặc B gây ra A, hoặc một số yếu tố khác gây ra cả A và B. Những giải thích này là gọi là "mối quan hệ nhân quả".

Chỉ riêng mối tương quan không phân biệt được giải thích nào là đúng, vì vậy nó không cho bạn biết giải thích nào đúng. Quy luật này thường được tóm tắt bằng câu "Tương quan không đồng nghĩa với quan hệ nhân quả,", nó có cả trang Wikipedia riêng. Vậy bạn có thể làm gì để cung cấp bằng chứng về quan hệ nhân quả?

• Thời gian sử dụng. Nếu A xảy ra trước B, thì A có thể gây ra B nhưng không thể ngược lại (tại ít nhất là theo cách hiểu thông thường của chúng ta về quan hệ nhân quả). Thứ tự của các sự kiện có thể giúp chúng ta suy luận hướng nhân quả, nhưng nó không loại trừ khả năng rằng một cái gì đó khác gây ra cả A và B.

• Sử dụng tính ngẫu nhiên. Nếu bạn chia một mẫu lớn thành hai nhóm một cách ngẫu nhiên và tính trung bình của hầu hết các biến, bạn mong đợi sự khác biệt sẽ nhỏ. Nếu các nhóm gần như giống hệt nhau trong tất cả các biến ngoại trừ một biến, bạn có thể loại bỏ các mối quan hệ giả.

Điều này hoạt động ngay cả khi bạn không biết các biến liên quan là gì, nhưng nó hoạt động tốt hơn nếu bạn biết quan hệ giữa các biến, bởi vì bạn có thể kiểm tra xem các nhóm có giống nhau không.

Những ý tưởng này là động lực cho thử nghiệm ngẫu nhiên kiểm soát, trong đó các đối tượng được chỉ định ngẫu nhiên vào hai (hoặc nhiều hơn) nhóm: một nhóm điều trị nhận được một số loại can thiệp, giống như một loại thuốc mới, và một nhóm kiểm soát không nhận được sự can thiệp quy ước, hoặc một phương pháp điều trị khác mà tác dụng của chúng đã được biết đến.

Một thử nghiệm kiểm soát ngẫu nhiên là cách tin cậy nhất để chứng minh mối quan hệ nhân quả, và đây là nền tảng của y học dựa trên khoa học.

Rất tiếc, các thử nghiệm có kiểm soát chỉ có thể thực hiện được trong khoa học phòng thí nghiệm, y học, và một vài ngành khác. Trong các ngành khoa học xã hội, các thí nghiệm kiểm soát hiếm khi được thực hiện, thường là vì chúng không thể hoặc không tuân thủ các chuẩn mực đạo đức.

Một giải pháp khác là tìm kiếm thí nghiệm tự nhiên, trong đó áp dụng các "phương trình" khác nhau cho các nhóm tương tự nhau. Một điều nguy hiểm của thí nghiệm tự nhiên là các nhóm có thể khác nhau theo những cách không rõ ràng.

Trong một số trường hợp, có thể suy luận các mối quan hệ nhân quả bằng cách sử dụng phân tích hồi quy, đây là chủ đề của Chương 11.

Bài tập

Sử dụng dữ liệu từ NSFG, tạo một biểu đồ phân tán so sánh trọng lượng sinh sản với tuổi của mẹ. Vẽ các phân vị của trọng lượng sinh sản so với tuổi của mẹ. Tính toán hệ số tương quan Pearson và Spearman. Bạn sẽ đặc tính được mối quan hệ giữa các biến như thế nào?

Bảng chú giải

Biểu đồ phân tán

Hình ảnh trực quan về mối quan hệ giữa hai biến, hiển thị một điểm cho mỗi hàng dữ liệu.

Jitter

Những điểm dữ liệu dao động ngẫu nhiên để phục vụ cho mục đích trực quan

Bão hòa

Sự mất thông tin khi nhiều điểm dữ liệu được vẽ chồng lên nhau.

Tương quan

Một thống kê đo lường sức mạnh của mối quan hệ giữa hai biến.

Tiêu chuẩn hóa

Để biến đổi một tập hợp các giá trị sao cho giá trị trung bình của chúng bằng 0 và phương sai bằng 1

Điểm chuẩn

Giá trị đã được tiêu chuẩn hóa để biểu thị theo đơn vị độ lệch chuẩn từ giá trị trung bình.

Hệ số hiệp phương sai

Một phép đo xu hướng của hai biến có xu hướng biến đổi cùng nhau.

Thứ hạng

Thứ hạng của một phần tử trong danh sách được sắp xếp.

Thử nghiệm ngẫu nhiên kiểm soát

Một thiết kế thử nghiệm trong đó các đối tượng được chia thành các nhóm một cách ngẫu nhiên, và các nhóm khác nhau được điều trị khác nhau.

Nhóm điều trị

Một nhóm trong thử nghiệm có kiểm soát nhận được một số hình thức can thiệp.

Nhóm kiểm soát

Một nhóm trong thử nghiệm kiểm soát không được nhận bất kỳ liệu trình nào, hoặc là được tiêm một liệu trình mà tác dụng của nó đã biết trước.

Thí nghiệm tự nhiên

Một thiết kế thử nghiệm tận dụng sự phân chia nhóm tự nhiên các đối tượng theo cách là gần như ngẫu nhiên.