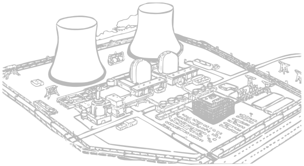


Regression analysis using Python

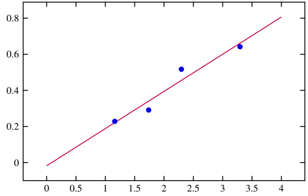
Eric Marsden

<eric.marsden@risk-engineering.org>

Some individuals use statistics as the drunken man uses lamp posts: for support rather than for illumination.

– attributed to Andrew Lang

Regression analysis



- ▷ Linear regression analysis means "fitting a straight line to data"
  - also called *linear modelling*
- ▷ It's a widely used technique to help **model** and **understand** real-world phenomena
  - easy to use
  - easy to understand intuitively
- ▷ Allows **prediction**

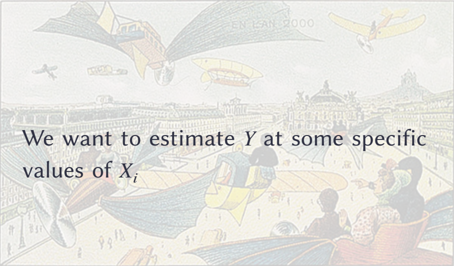
RISK ENGINEERING

Regression analysis

- ▷ A regression problem is composed of
  - an *outcome* or *response variable* Y
  - a number of *risk factors* or *predictor variables* X_i that affect Y
    - also called *explanatory variables*, or *features* in the machine learning community
  - a question about Y, such as *How to predict* Y *under different conditions?*
- ▷ Y is sometimes called the *dependent variable* and X_i the *independent variables*
  - not the same meaning as *statistical independence*
  - experimental setting where the X_i variables can be modified and changes in Y can be observed

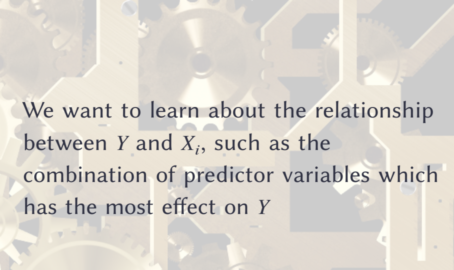
Regression analysis: objectives

Prediction



We want to estimate Y at some specific values of X_i

Model inference



We want to learn about the relationship between Y and X_i, such as the combination of predictor variables which has the most effect on Y

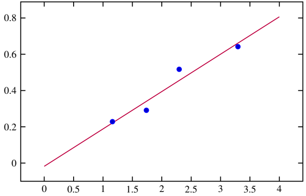
Univariate linear regression

(when all you have is a single predictor variable)

Linear regression

- Linear regression: one of the simplest and most commonly used statistical modeling techniques

- Makes strong assumptions about the relationship between the predictor variables (X_i) and the response (Y)
  - (a linear relationship, a straight line when plotted)
  - only valid for *continuous* outcome variables (not applicable to category outcomes such as *success*/*failure*)

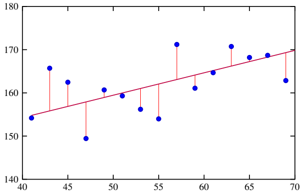


$$\underbrace{y}_{\text{outcome variable}} = \underbrace{\beta_0}_{\text{intercept}} + \underbrace{\beta_1}_{\text{slope}} \times \underbrace{x}_{\text{predictor variable}} + \text{error}$$

"Fitting a line through data"

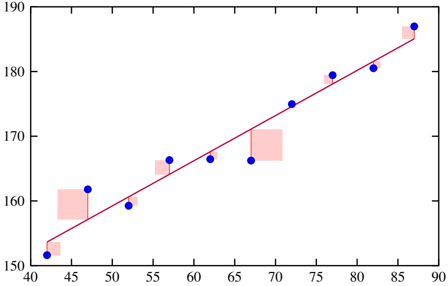
Linear regression

- Assumption: $y = \beta_0 + \beta_1 \times x + \text{error}$
- Our task: estimate β_0 and β_1 based on the available data
- Resulting model is $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 \times x$
  - the "hats" on the variables represent the fact that they are **estimated from the available data**
  - $\hat{y}$ is read as *"the estimator for y"*
- β_0 and β_1 are called the model *parameters* or *coefficients*
- **Objective**: minimize the *error*, the difference between our observations and the predictions made by our linear model
  - minimize the length of the red lines in the figure to the right (called the "residuals")



Ordinary Least Squares regression

- ▷ Ordinary Least-Squares (OLS) regression: a method for selecting the model parameters
  - β_0 and β_1 are chosen to minimize the **square of the distance** between the predicted values and the actual values
  - equivalent to minimizing the size of the red rectangles in the figure to the right
- ▷ An application of a *quadratic loss function*
  - in statistics and optimization theory, a *loss function*, or *cost function*, maps from an observation or event to a number that represents some form of "cost"



Simple linear regression: example

- The *British Doctors' Study* followed the health of a large number of physicians in the UK over the period 1951–2001
- Provided conclusive evidence of linkage between smoking and lung cancer, myocardial infarction, respiratory disease and other illnesses
- Provides data on annual mortality for a variety of diseases at four levels of cigarette smoking:
  1. never smoked
  2. 1-14 per day
  3. 15-24 per day
  4. > 25 per day

More information: `ctsu.ox.ac.uk/research/british-doctors-study`

RISK ENGINEERING

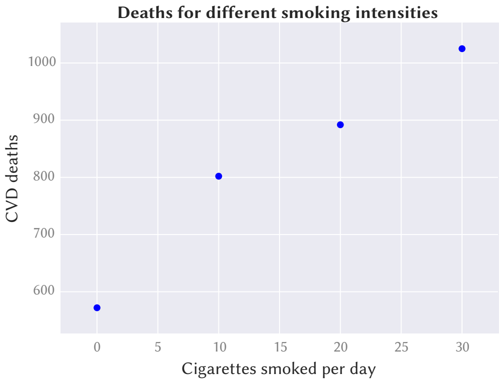
Simple linear regression: the data

cigarettes smoked (per day)	CVD mortality (per 100 000 men per year)	lung cancer mortality (per 100 000 men per year)
0	572	14
10 (actually 1-14)	802	105
20 (actually 15-24)	892	208
30 (actually >24)	1025	355

CVD: cardiovascular disease

Source: British Doctors' Study

Simple linear regression: plots

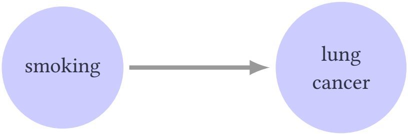



```python
import pandas
import matplotlib.pyplot as plt

data = pandas.DataFrame({"cigarettes": [0,10,20,30],
                         "CVD": [572,802,892,1025],
                         "lung": [14,105,208,355]});
data.plot("cigarettes", "CVD", kind="scatter")
plt.title("Deaths for different smoking intensities")
plt.xlabel("Cigarettes smoked per day")
plt.ylabel("CVD deaths")
```

Quite tempting to conclude that cardiovascular disease deaths increase linearly with cigarette consumption…

Aside: beware assumptions of causality

1964: the US Surgeon General issues a report claiming that cigarette smoking causes lung cancer, based mostly on correlation data similar to the previous slide.

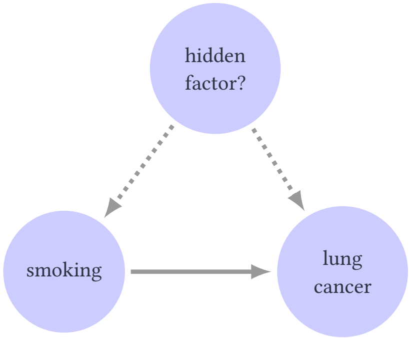


RISK ENGINEERING

Aside: beware assumptions of causality

1964: the US Surgeon General issues a report claiming that cigarette smoking causes lung cancer, based mostly on correlation data similar to the previous slide.

However, correlation is not sufficient to demonstrate causality. There might be some hidden genetic factor that causes both lung cancer and desire for nicotine.



Beware assumptions of causality

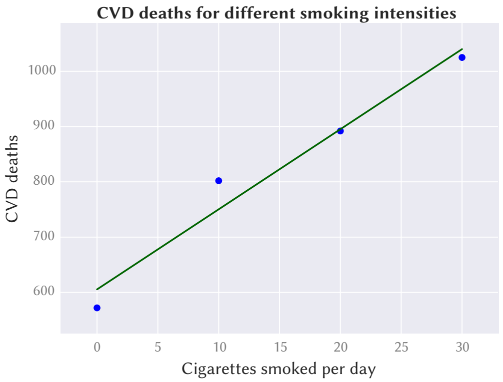
- To demonstrate the causality, you need a **randomized controlled experiment**
- Assume we have the power to force people to smoke or not smoke
  - and ignore moral issues for now!
- Take a large group of people and divide them into two groups
  - one group is obliged to smoke
  - other group not allowed to smoke (the “control” group)
- Observe whether smoker group develops more lung cancer than the control group
- We have eliminated any possible hidden factor causing both smoking and lung cancer
- More information: read about **design of experiments**

Fitting a linear model in Python

- In these examples, we use the `statsmodels` library for statistics in Python
  - other possibility: the `scikit-learn` library for machine learning
- We use the formula interface to OLS regression, in `statsmodels.formula.api`
- Formulas are written `outcome ~ observation`
  - meaning "build a linear model that predicts variable *outcome* as a function of input data on variable *observation*"

RISK ENGINEERING

Fitting a linear model

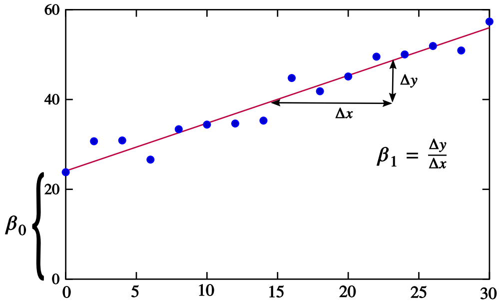



```python
import numpy, pandas
import matplotlib.pyplot as plt
import statsmodels.formula.api as smf

df = pandas.DataFrame({"cigarettes": [0,10,20,30],
                       "CVD": [572,802,892,1025],
                       "lung": [14,105,208,355]});
df.plot("cigarettes", "CVD", kind="scatter")
lm = smf.ols("CVD ~ cigarettes", data=df).fit()
xmin = df.cigarettes.min()
xmax = df.cigarettes.max()
X = numpy.linspace(xmin, xmax, 100)
# params[0] is the intercept (beta₀)
# params[1] is the slope (beta₁)
Y = lm.params[0] + lm.params[1] * X
plt.plot(X, Y, color="darkgreen")
```

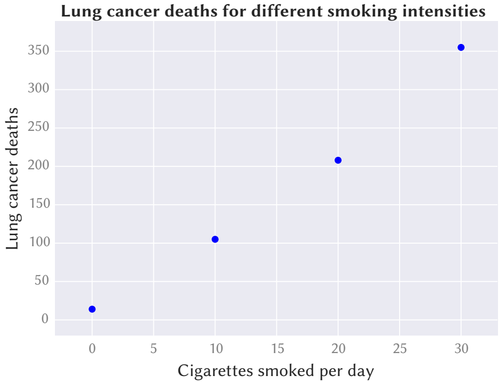
RISK ENGINEERING

Parameters of the linear model

- β_0 is the **intercept** of the regression line (where it meets the $X = 0$ axis)
- β_1 is the **slope** of the regression line
- Interpretation of $\beta_1 = 0.0475$: a "unit" increase in cigarette smoking is associated with a 0.0475 "unit" increase in deaths from lung cancer



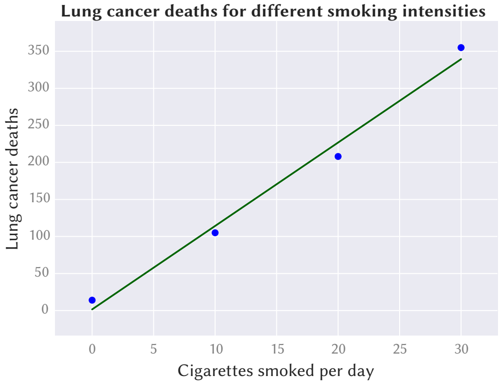
Scatterplot of lung cancer deaths




```python
import pandas
import matplotlib.pyplot as plt

data = pandas.DataFrame({"cigarettes": [0,10,20,30],
                         "CVD": [572,802,892,1025],
                         "lung": [14,105,208,355]});
data.plot("cigarettes", "lung", kind="scatter")
plt.xlabel("Cigarettes smoked per day")
plt.ylabel("Lung cancer deaths")
```

Quite tempting to conclude that lung cancer deaths increase linearly with cigarette consumption…

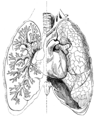
Fitting a linear model




```python
import numpy, pandas
import matplotlib.pyplot as plt
import statsmodels.formula.api as smf

df = pandas.DataFrame({"cigarettes": [0,10,20,30],
                       "CVD": [572,802,892,1025],
                       "lung": [14,105,208,355]});
df.plot("cigarettes", "lung", kind="scatter")
lm = smf.ols("lung ~ cigarettes", data=df).fit()
xmin = df.cigarettes.min()
xmax = df.cigarettes.max()
X = numpy.linspace(xmin, xmax, 100)
# params[0] is the intercept (beta₀)
# params[1] is the slope (beta₁)
Y = lm.params[0] + lm.params[1] * X
plt.plot(X, Y, color="darkgreen")
```

Download the associated Python notebook at risk-engineering.org

Using the model for prediction



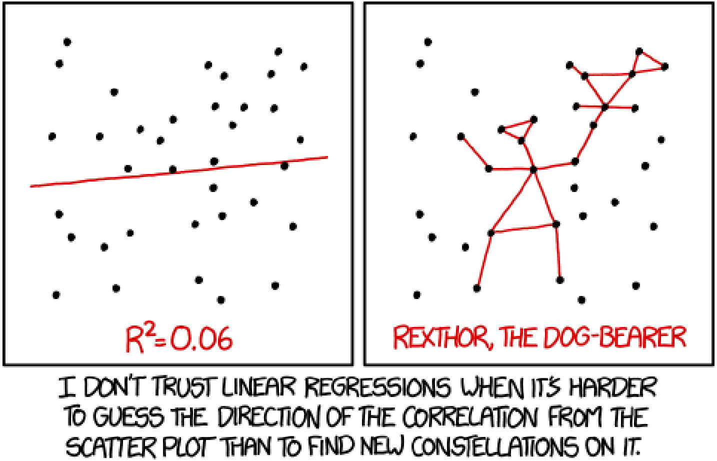
Q: *What is the expected lung cancer mortality risk for a group of people who smoke 15 cigarettes per day?*

```python
import numpy, pandas
import statsmodels.formula.api as smf

df = pandas.DataFrame({"cigarettes": [0,10,20,30],
                       "CVD": [572,802,892,1025],
                       "lung": [14,105,208,355]});
# create and fit the linear model
lm = smf.ols(formula="lung ~ cigarettes", data=df).fit()
# use the fitted model for prediction
lm.predict({"cigarettes": [15]}) / 100000.0
# probability of mortality from lung cancer, per person per year
array([ 0.001705])
```

Assessing model quality

- ▷ How do we assess how well the linear model fits our observations?
  - make a visual check on a scatterplot
  - use a quantitative measure of "goodness of fit"
- ▷ **Coefficient of determination** r^2: a number that indicates how well data fit a statistical model
  - it's the proportion of total variation of outcomes explained by the model
  - $r^2 = 1$: regression line fits perfectly
  - $r^2 = 0$: regression line does not fit at all
- ▷ For simple linear regression, r^2 is simply the square of the sample correlation coefficient r

Assessing model quality

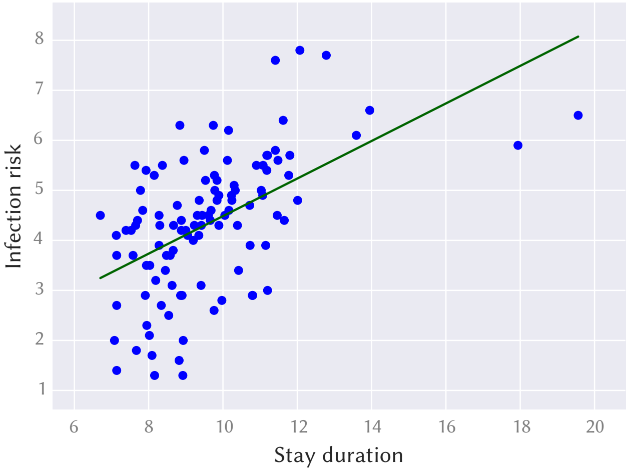


Source: xkcd.com/1725, CC BY-NC licence

Information on the linear model

```
> lm = smf.ols(formula='lung ~ cigarettes', data=df).fit()
> lm.summary()
                            OLS Regression Results
==============================================================================
Dep. Variable:                   lung   R-squared:                       0.987
Model:                            OLS   Adj. R-squared:                  0.980
Method:                 Least Squares   F-statistic:                     151.8
Date:                Wed, 06 Jan 2016   Prob (F-statistic):            0.00652
Time:                        14:01:34   Log-Likelihood:                -16.359
No. Observations:                   4   AIC:                             36.72
Df Residuals:                       2   BIC:                             35.49
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      1.6000     17.097      0.094      0.934       -71.964    75.164
cigarettes    11.2600      0.914     12.321      0.007         7.328    15.192
==============================================================================
Omnibus:                          nan   Durbin-Watson:                   2.086
Prob(Omnibus):                    nan   Jarque-Bera (JB):                0.534
Skew:                          -0.143   Prob(JB):                        0.766
Kurtosis:                       1.233   Cond. No.                         31.4
==============================================================================
```

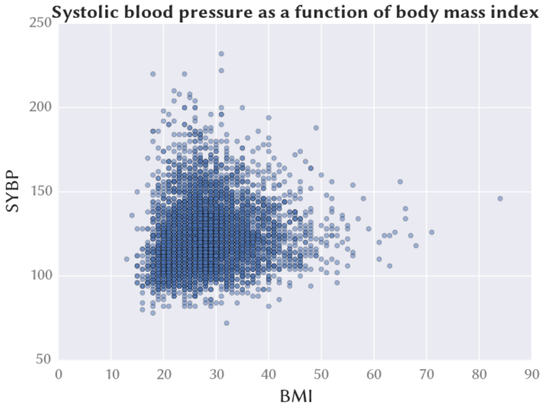
Example: nosocomial infection risk



Longer stays in hospitals are associated with a higher risk of nosocomial infection

Data source: SENIC survey on nosocomial risk

Example: blood pressure and BMI



Data on Body Mass Index and systolic blood pressure

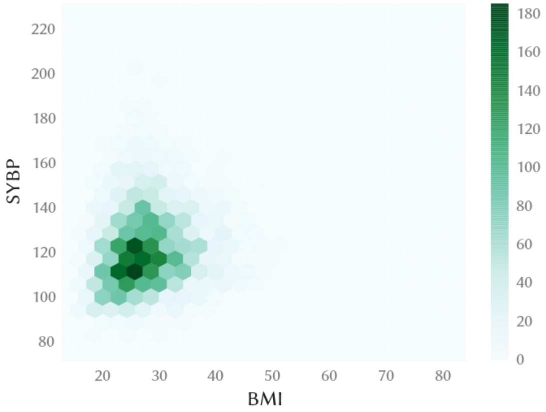
A higher body mass index is correlated with higher blood pressure

Python with a Pandas dataframe:

```
df.plot(x="BMI", y="SYBP",
        kind="scatter", alpha=0.5)
```

Data source: NHANES survey, US CDC, 2009–2010

Example: blood pressure and BMI



Data on Body Mass Index and systolic blood pressure

A higher body mass index is correlated with higher blood pressure

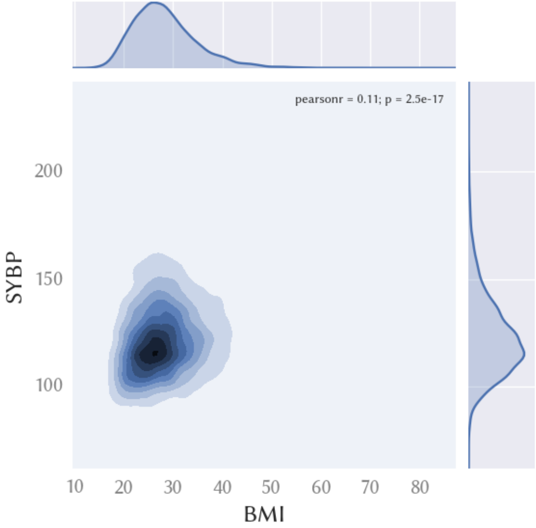
Same data as previous slide, with a "hexplot" instead of scatterplot

Python with a Pandas dataframe:

```
df.plot(x="BMI", y="SYBP",
        kind="hexbin", gridsize=25)
```

Data source: NHANES survey, US CDC, 2009–2010

Example: blood pressure and BMI



Data on Body Mass Index and systolic blood pressure. A higher body mass index is correlated with higher blood pressure.

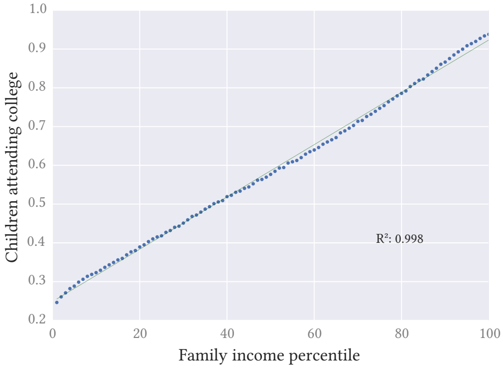
Same data as previous slide, with a kernel density plot instead of scatterplot.

Python with a Pandas dataframe using the Seaborn library:

```
jointplot(data=df,
          x="BMI", y="SYBP",
          kind="kde")
```

Data source: NHANES survey, US CDC, 2009–2010

Example: intergenerational mobility in the USA



Percentage of children in college at age 19 plotted against the percentile rank of their parents' income. Data for the USA.

Intergenerational mobility (for example chance of moving from bottom to top fifth of income distribution) is similar for children entering labor market today than in the 1970s. However, level of inequality has diminished, so consequences of the "birth lottery" are greater today.

(Political and moral implications of this analysis, and associated risks, are beyond the scope of these slides, but are one of our motivations for making these materials available for free…)

→ scholar.harvard.edu/hendren/publications/united-states-still-land-opportunity-recent-trends-intergenerational-mobility

Data source: opportunityinsights.org/

Exercise: the "Dead grandmother problem"

Problem. Research by Prof. M. Adams suggests that the week prior to exam time is an extremely dangerous time for the relatives of university students. Data shows that a student's grandmother is far more likely to die suddenly just before the student takes an exam, than at any other time of year.

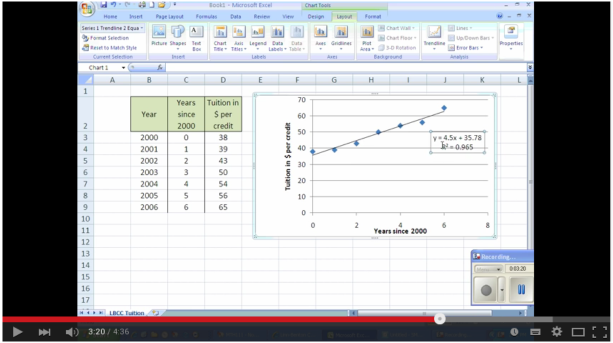
Theory. Family members literally worry themselves to death over the outcome of their relatives' performance on each exam.

Task: use linear regression to confirm that the severity of this phenomenon is correlated to the student's current grade.

Data source: `math.toronto.edu/mpugh/DeadGrandmother.pdf`

RISK ENGINEERING

Aside: linear regression in Excel



Summary:

- ▷ Functions `SLOPE` and `INTERCEPT`
- ▷ Correlation coefficient: function `CORREL`

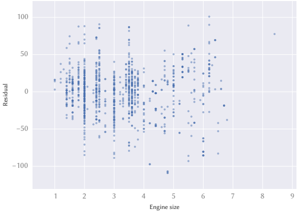
Explanatory video: `youtu.be/ExfknNCvBYg`

Residuals plot

- In linear regression, the residual data is the difference between the observed data of the outcome variable y and the predicted values $\hat{y}$

$$residual = y - \hat{y}$$

- The residuals plot should look "random" (no discernible pattern)
  - if the residuals are not random, they suggest that your model is systematically incorrect, meaning it can be improved
  - see example to the right with no specific pattern
- If you spot a trend in the residuals plot (increasing, decreasing, "U" shape), the data is most likely non-linear
  - so a linear model is not a good choice for this problem...



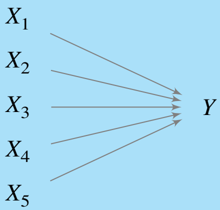
Multivariate regression

What is multivariate linear regression?

Univariate linear regression

$$X \longrightarrow Y$$

Multivariate linear regression



Multivariate linear regression involves **more than one predictor** variable

Multivariate linear regression: equations

- Recall the equation for univariate linear regression:

$$\hat{y} = \beta_0 + \beta_1 x$$

- Equation for **multivariate linear regression**:

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n$$

- The outcome variable is assumed to be a linear combination of the predictor variables (the inputs)

RISK ENGINEERING

Example: prediction using a multivariate dataset

- **Objective**: predict energy output at a Combined Cycle Power Plant
- Data available: hourly averages of variables

Meaning	Name	Range
Ambient Temperature	AT	1.81 – 37.11˚C
Ambient Pressure	AP	992.89 – 1033.30 millibar
Relative Humidity	RH	25.56% – 100.16%
Exhaust Vacuum	V	25.36 – 81.56 cm Hg
Net hourly electrical energy output	PE	420.26 – 495.76 MW

- Let's try to build a multivariate linear model to predict PE given inputs AT, AP, RH and V

Dataset: `archive.ics.uci.edu/ml/datasets/Combined+Cycle+Power+Plant`

RISK ENGINEERING

Example: prediction using a multivariate dataset

- Dataset contains 9568 data points collected from a combined cycle power plant over 6 years, when power plant was under full load
- A combined cycle power plant is composed of gas turbines, steam turbines and heat recovery steam generators
  - electricity is generated by gas & steam turbines, which are combined in one cycle
  - three ambient variables affect performance of the gas turbine
  - exhaust vacuum affects performance of the steam turbine
- Data consists of hourly averages taken from various sensors located around the plant that record the ambient variables every second
- Let's load it into Python and examine it using the `pandas` library

Dataset: `archive.ics.uci.edu/ml/datasets/Combined+Cycle+Power+Plant`

Example: prediction using a multivariate dataset

```
> import pandas
> data = pandas.read_csv("data/CCPP.csv")
> data.head()
      AT      V       AP     RH      PE
0  14.96  41.76  1024.07  73.17  463.26
1  25.18  62.96  1020.04  59.08  444.37
2   5.11  39.40  1012.16  92.14  488.56
3  20.86  57.32  1010.24  76.64  446.48
4  10.82  37.50  1009.23  96.62  473.90
> data.describe()
                AT           V           AP           RH           PE
count  9568.000000  9568.000000  9568.000000  9568.000000  9568.000000
mean     19.651231    54.305804  1013.259078    73.308978   454.365009
std       7.452473    12.707893     5.938784    14.600269    17.066995
min       1.810000    25.360000   992.890000    25.560000   420.260000
25%      13.510000    41.740000  1009.100000    63.327500   439.750000
50%      20.345000    52.080000  1012.940000    74.975000   451.550000
75%      25.720000    66.540000  1017.260000    84.830000   468.430000
max      37.110000    81.560000  1033.300000   100.160000   495.760000
```

Dataset: archive.ics.uci.edu/ml/datasets/Combined+Cycle+Power+Plant

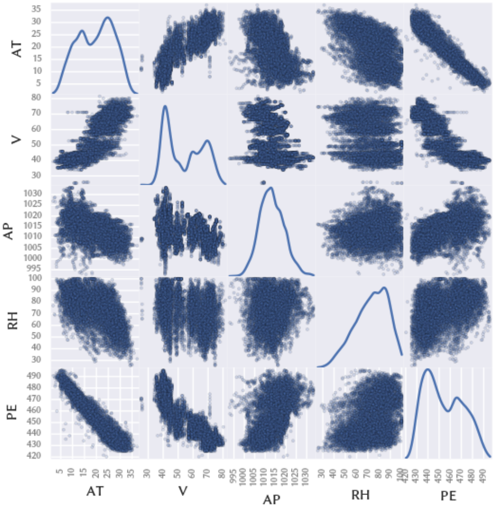
RISK ENGINEERING

Visualizing multivariate data: scatterplot matrix

We can obtain a first impression of the dependency between variables by examining a multidimensional scatterplot

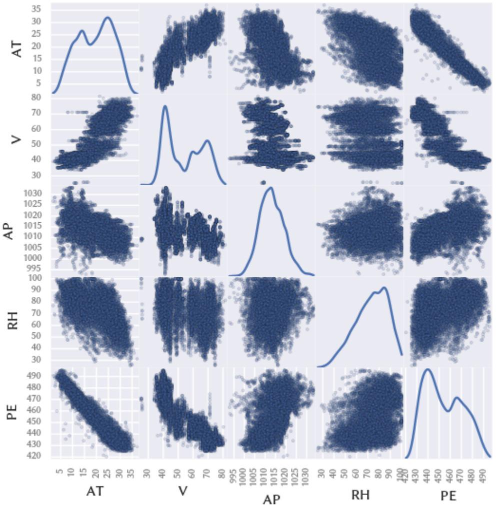
```python
from pandas.tools.plotting import scatter_matrix
data = pandas.read_csv("data/CCPP.csv")
scatter_matrix(data, diagonal="kde")
```

In this matrix, the diagonal contains a plot of the distribution of each variable.



RISK ENGINEERING

Interpreting the scatterplot matrix



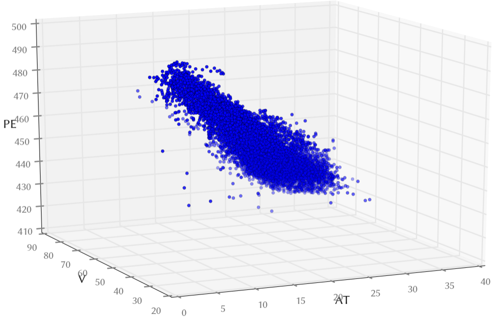
Observations:

- ▷ approximately linear relationship between PE and the negative of AT
- ▷ approximately linear relationship between PE and negative of V

Visualizing multivariate data: 3D plotting

It is sometimes useful to examine 3D plots of your observations

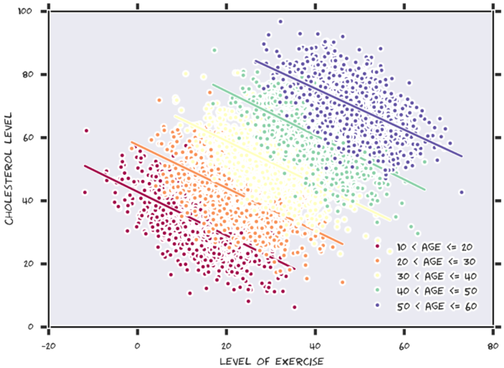
```
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt

fig = plt.figure(figsize=(12, 8))
ax = Axes3D(fig, azim=-115, elev=15)
ax.scatter(data["AT"], data["V"], data["PE"])
ax.set_xlabel("AT")
ax.set_ylabel("V")
ax.set_zlabel("PE")
```



Importance of preliminary data analysis

Consider a study that measures weekly exercise and cholesterol in various age groups.

If we plot exercise against cholesterol and segregate by age, we see a downward trend in each group: more exercise leads to lower cholesterol.



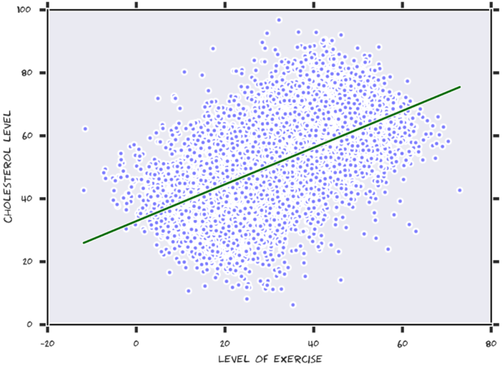
Note: fake (but plausible!) data

RISK ENGINEERING

Importance of preliminary data analysis

If we don't segregate by age, we get the plot to the right, which could lead to an incorrect conclusion that more exercise is correlated with more cholesterol.

There is an underlying variable age: older people tend to exercise more, and also have higher cholesterol.



This example from *Causal inference in statistics: a primer*, J. Pearl et al, 2016

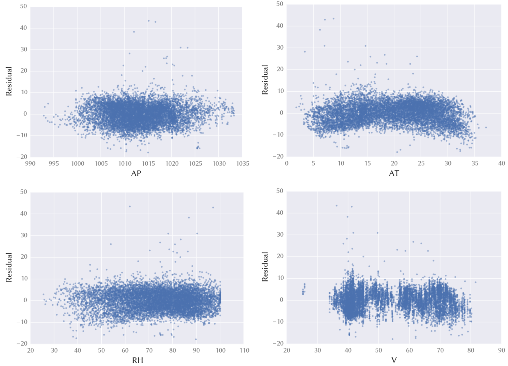
CCPP example: least squares regression with Python

```
# create fitted model using “formula” API of the statsmodels library
import statsmodels.formula.api as smf
> lm = smf.ols(formula='PE ~ AT + V + AP + RH', data=data).fit()
> print(lm.params)
Intercept    451.067793
AT            -1.974731
V             -0.234992
AP             0.065540
RH            -0.157598
```

This means that the best formula to estimate output power as a function of AT, V, AP and RH is

$$PE = 451.067793 - 1.974731\,AT - 0.234992\,V + 0.065540\,AP - 0.157598\,RH$$

Residuals plots



The residuals for each predictor variable look random, except for a mild quadratic shape for AT, which we will ignore here.

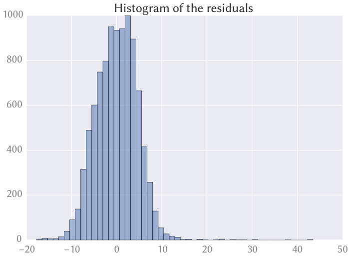
Residuals histogram

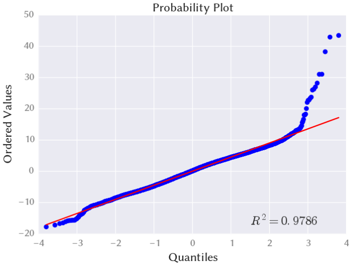
One assumption underlying linear regression is that the variance of the residuals is normally distributed (follows a Gaussian distribution).

Can be checked by plotting a histogram or a Q-Q plot of the residuals, as shown to the right.

Example to the right: we have a deviation from normality for large prediction errors, but overall residuals follow a normal distribution.

Download the associated Python notebook at `risk-engineering.org`





CCPP example: prediction

Assuming the values below for our input variables, what is the predicted output power?

AT	9.48
V	44.71
AP	1019.12
RH	66.43

```
> m = pandas.DataFrame({"AT": [9.48], "V": [44.71],
      "AP": [1019.12], "RH": [66.43]})
> lm.predict(m)
[ 478.25471442]
```

Conclusion: the predicted output power is 478.3 MW.

Assessing goodness of fit: R^2

- For multiple linear regression, the coefficient of determination R^2 is calculated as

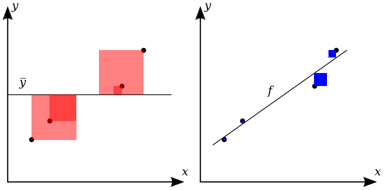
$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

  where

  - $SS_{res} = \sum_i (y_i - \widehat{y_i})^2$ is the sum of the square of the residuals
  - $SS_{tot} = \sum_i (y_i - \bar{y})^2$ is the total sum of squares
  - y_i are the observations, for $i = 1 \ldots n$
  - $\widehat{y_i}$ are the predictions, for $i = 1 \ldots n$
  - $\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i$ is the mean of the observations

- The better the fit, the closer R^2 is to 1

- R^2 measures the **proportion of variance** in the observed data that is **explained by the model**



Areas of red squares: squared residuals with respect to the average value

Areas of blue squares: squared residuals with respect to the linear regression

Determining R^2 in Python

```
> lm.summary()
                            OLS Regression Results
==============================================================================
Dep. Variable:                     PE   R-squared:                       0.927
Model:                            OLS   Adj. R-squared:                  0.927
Method:                 Least Squares   F-statistic:                 2.295e+04
Date:                Tue, 05 Jan 2016   Prob (F-statistic):               0.00
Time:                        17:21:31   Log-Likelihood:                -21166.
No. Observations:                7196   AIC:                         4.234e+04
Df Residuals:                    7191   BIC:                         4.238e+04
Df Model:                           4
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept    460.9650     11.308     40.764      0.000       438.798   483.132
AT            -1.9809      0.018   -111.660      0.000        -2.016    -1.946
V             -0.2303      0.008    -27.313      0.000        -0.247    -0.214
AP             0.0556      0.011      5.073      0.000         0.034     0.077
RH            -0.1576      0.005    -32.827      0.000        -0.167    -0.148
==============================================================================
Omnibus:                      864.810   Durbin-Watson:                   2.009
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             4576.233
Skew:                          -0.459   Prob(JB):                         0.00
Kurtosis:                       6.797   Cond. No.                     2.13e+05
==============================================================================
```

Warnings concerning linear regression

Warnings concerning use of linear regression

1. Check that your data is really linear!
2. Make sure your sample size is sufficient
3. Don't use a regression model to predict responses outside the range of data that was used to build the model
4. Results can be highly sensitive to treatment of **outliers**
5. Multiple regression: check that your predictors are independent
6. Beware order of effect problems
   - regression shows correlation but does not necessarily imply causality
7. Beware the **regression to the mean** effect

RISK ENGINEERING

⚠ Check assumptions underlying linear regression

- ▷ Examine scatterplot of outcome variable with each predictor to validate the assumption of linearity
- ▷ Other assumptions underlying the use of linear regression:
  - Check that the mean of the residuals is almost equal to zero for each value of outcome
  - Check that the residuals have constant variance (→ residuals scatterplot on slide 24)
  - Check that residuals are uncorrelated (→ residuals scatterplot)
  - Check that residuals are normally distributed (→ residuals histogram or QQ-plot) or that you have an adequate sample size to rely on large sample theory

⚠ Make sure your sample size is sufficient

▷ There are no rules on required sample size for a regression analysis

- depends on the number of predictor variables, on the effect size, the objective of the analysis

▷ Some general observations:

- bigger samples are better (give more confidence in the model)
- sample size is often determined by pragmatic considerations (measurements may be expensive, limited historical data available)
- sample size should be seen as one consideration in an optimization problem where the cost in time/money/effort of obtaining more data is weighed against the benefits (better predictions, improved understanding)

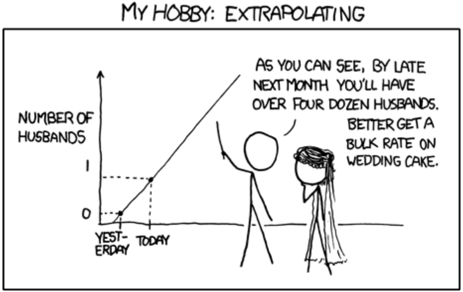
RISK ENGINEERING

⚠ Extrapolate with care

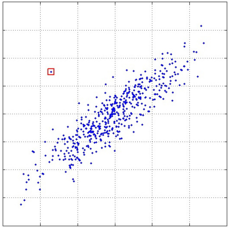
To infinity and beyond!

When using a linear model for prediction, be very careful when predicting responses outside of the range of data that was used to build the model.

Make sure you have well-grounded scientific reasons for arguing that the model also applies in areas where you don't have available data.



⚠ Treatment of outlier data



- ▷ Real datasets often contain **spurious data points**
  - errors made in measurement, noise, data entry errors…
- ▷ These may have a significant impact on your predictions
- ▷ However, some outlier data may just be "different" but meaningful observations
  - possibly an early warning sign of an upcoming catastrophe!
- ▷ The best method of handling outliers depends on the objective of your analysis, on how you obtained your data…

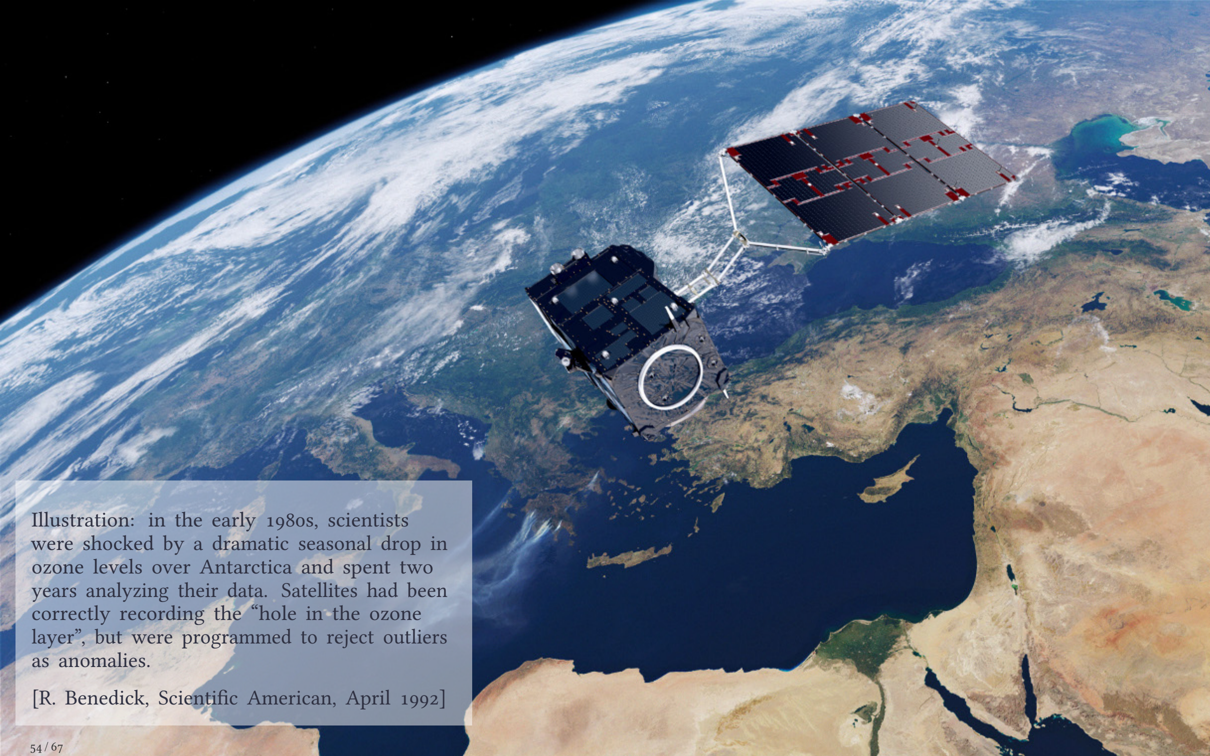


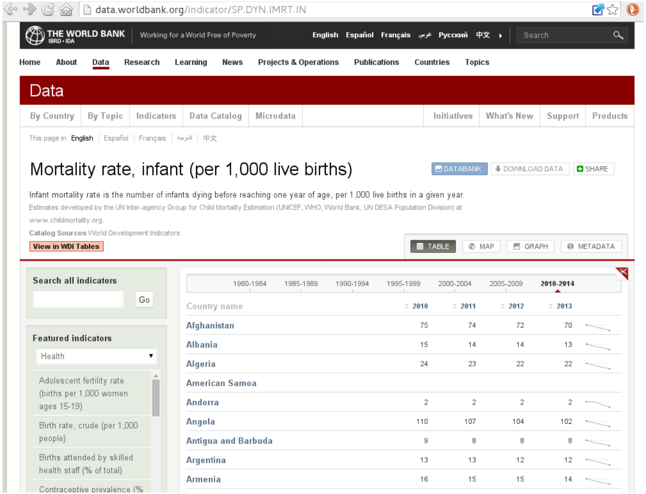
Illustration: in the early 1980s, scientists were shocked by a dramatic seasonal drop in ozone levels over Antarctica and spent two years analyzing their data. Satellites had been correctly recording the “hole in the ozone layer”, but were programmed to reject outliers as anomalies.

[R. Benedick, Scientific American, April 1992]

⚠ Recommendations for handling outliers

- Analyze outliers individually, for instance by plotting your data
- Eliminate from the dataset any outliers that you are confident you can identify as being the result of errors in measurement or data entry
- For remaining outliers, report prediction results both with and without the outliers
- Consider using a *robust* linear regression technique
  - example: `RLM` from the `statsmodels` library (Python)
  - `RANSAC` from the `scikit-learn` library (Python)

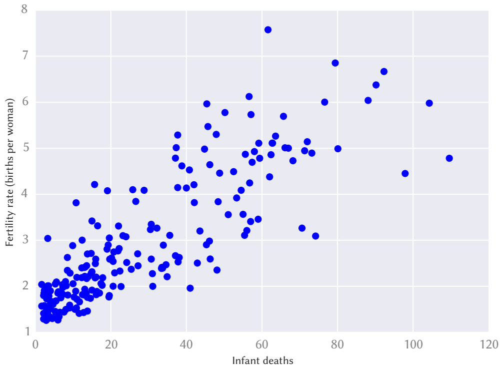
⚠ Beware of order of effect problems



Consider infant mortality data from the World Bank

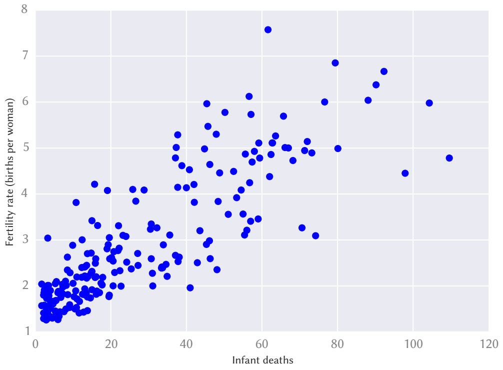
Data source: `data.worldbank.org`

Predictor and outcome variables



- ▷ Two variables:
  - infant mortality rate (per 1000 births)
  - number of births per woman
- ▷ Which is the predictor variable and which is the outcome?

Predictor and outcome variables



- ▷ Two variables:
  - infant mortality rate (per 1000 births)
  - number of births per woman
- ▷ Which is the predictor variable and which is the outcome?
- ▷ Choice 1: fertility = f(infant-mortality)
  - predictor: infant mortality rate
  - outcome: births per woman
- ▷ Choice 2: infant-mortality = f(fertility)
  - predictor: births per woman
  - outcome: infant mortality rate

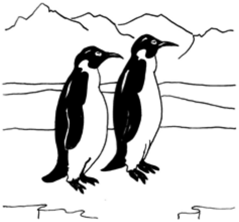
Data source: `data.worldbank.org`

Predictor and outcome variables

- ▷ The answer depends on the **framing of the research question**
- ▷ If hypothesis is influence of infant mortality on number of births per woman, then
  - predictor: infant mortality rate
  - outcome: births per woman
- ▷ If hypothesis is influence of number of births per woman on infant mortality, then
  - predictor: births per woman
  - outcome: infant mortality rate

RISK ENGINEERING

⚠ Directionality of effect problem



"Do you think all these film crews brought on global warming or did global warming bring on all these film crews?"

Examples:

- ▷ People who exercise more tend to have better health
- ▷ Police departments with higher budgets tend to be located in areas with high crime levels
- ▷ Middle-aged men who wear hats are more likely to be bald
- ▷ Young smokers who buy contraband cigarettes tend to smoke more

RISK ENGINEERING

⚠ Regression to the mean

- ▷ Following an extreme random event, the next random event is likely to be less extreme
  - if a variable is extreme on its first measurement, it will tend to be closer to the average on its second measurement
- ▷ Examples:
  - If today is extremely hot, you should probably expect tomorrow to be hot, but not quite as hot as today
  - If a baseball player just had by far the best season of his career, his next year is likely to be a disappointment
- ▷ Extreme events tend to be followed by something closer to the norm

⚠ Regression to the mean

> *Statistical regression to the mean predicts that patients selected for abnormalcy will, on the average, tend to improve. We argue that most improvements attributed to the placebo effect are actually instances of statistical regression.*
>
> *Thus, we urge caution in interpreting patient improvements as causal effects of our actions and should avoid the conceit of assuming that our personal presence has strong healing powers. [McDonald et al 1983]*

- Group of patients that are treated with a placebo are affected by two processes:
  - genuine psychosomatic placebo effect
  - “get better anyway” effect (regression to the mean)

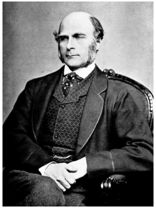
Source: *How much of the placebo ‘effect’ is really statistical regression?*, C. McDonald & S. Mazzuca, *Statistics in medicine*, vol. 2, 417–427 (1983)

⚠ Regression to the mean

- ▷ Classical example of regression to the mean: **effectiveness of speed cameras** in preventing accidents
- ▷ Speed cameras tend to be installed after an exceptional series of accidents at that location
- ▷ If the accident rate is particularly high somewhere one year, it will probably be lower the next year
  - irrespective of whether a speed camera is installed…
- ▷ To avoid this bias, implement a *randomized trial*
  - choose several similar sites
  - allocate them at random to have a camera or no camera
  - check whether the speed camera has a statistically measurable effect

RISK ENGINEERING

Aside: origin of the term



- Francis Galton (1822–1911) was an English anthropologist and statistician (and polymath)
- Discovered/formalized the statistical concept of correlation
- Collected data on the height of the descendants of extremely tall and extremely short trees
  - to analyze how "co-related" trees were to their parents
  - publication: *Regression Towards Mediocrity in Hereditary Stature* (1866)

> *It appeared from these experiments that the offspring did not tend to resemble their parents seeds in size, but to be always more mediocre than they - to be smaller than the parents, if the parents were large; to be larger than the parents, if the parents were small.*

- But towards the end of his life, studied whether human ability was hereditary and promoted eugenics…

Other applicable techniques

- ▷ Linear regression techniques are not applicable for category data, such as success/failure data
  - use *generalized linear models* (GLM) instead
- ▷ Sometimes **machine learning** algorithms can be more appropriate than regression techniques
  - example algorithms: random forest, support vector machines, neural networks

Image credits

- *L'avenue de l'Opéra* on slide 4 by Villemard, 1910 (BNF collection)
- Clockwork on slide 4: `flic.kr/p/edA7aA`, CC BY licence
- Heart on slide 10: Wikimedia Commons, public domain
- Lungs on slide 13: Wikimedia Commons, public domain
- Grandmother on slide 28: Marjan Lazarevski via `flic.kr/p/dJfAWQ`, CC BY-ND licence
- Coefficient of determination (slide 39): Orzetto via Wikimedia Commons, CC BY-SA licence
- Sentinel satellite on slide 54: copyright ESA/ATG medialab, ESA standard licence
- Speed camera on slide 56: Mick Baker via `flic.kr/p/bsBt8f`, CC BY-ND licence
- Photo of Francis Galton on slide 56: Wikimedia Commons, public domain

Further reading

- The Stanford Online class on Statistical Learning introduces supervised learning with a focus on regression and classification methods
  - → `online.stanford.edu`
- The online, open-access textbook *Forecasting: principles and practice*
  - → `otexts.org/fpp2` (uses R rather than Python)
- Online book *Practical regression and Anova using R*
  - → `cran.r-project.org/doc/contrib/Faraway-PRA.pdf`

Feedback welcome!

Was some of the content unclear? Which parts were most useful to you? Your comments to `feedback@risk-engineering.org` (email) or `@LearnRiskEng` (Twitter) will help us to improve these materials. Thanks!

`@LearnRiskEng`

`fb.me/RiskEngineering`

For more free content on risk engineering, visit `risk-engineering.org`