Probabilistic Linear regression

Giorgio Corani

Bayesian Data Analysis and Probabilistic Programming

giorgio.corani@supsi.ch

Based on..

- Chapter 3 of O. Martin, *Bayesian Analysis with Python, Second Edition*.
- Chapter 8 of *the Bayes rule book*
  https://www.bayesrulesbook.com/chapter-10.html
- Notebook by G. Corani

Linear regression

- We want to predict the value of Y given the observation of X.

- X and Y are random variables:
  - Y is the *dependent* (or *response*) variable
  - X is the *independent* variable (or *explanatory variable* or *covariate*)

Simple linear regression: a single explanatory variable

$$Y = \alpha + \beta X + \epsilon$$

- α (*intercept*): predicted value of Y for X = 0; it is a constant which calibrates the shift along the y-axis.
- β (*slope*): predicted change in Y for a unit change in X.
- ϵ is a noise which the scatters the observations around the line.

Simple linear regression

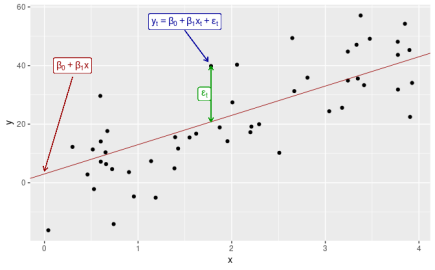

Figure 1: Linear regression

- The noise ϵ implies a deviation from the linear model. It captures anything that may affect Y other than X.
- We assume $\epsilon \sim N(0, \sigma^2)$.

The effect of σ

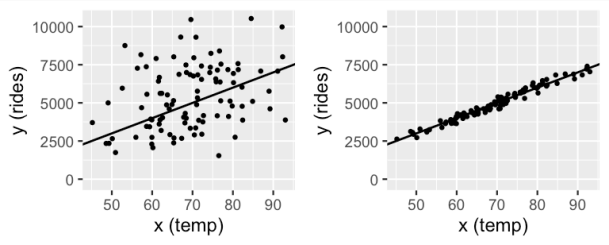

Figure 2: Effect of σ

- σ is the std dev of the noise ϵ.
- Large σ: large variability of the observations around the linear model, weak relationship.
- Small σ: the observations deviates little from the model; strong relationship.

The effect of σ

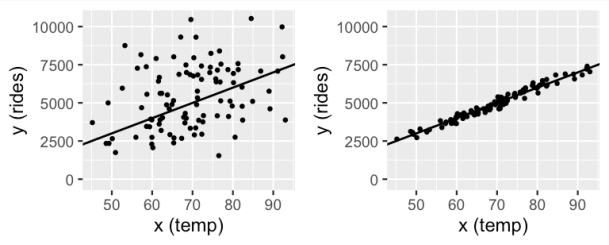

Figure 3: Effect of σ

- Under the normal assumption:
  - about 68% of the observations lie in an interval of $\pm 1\sigma$ around the regression line.
  - how many observations will lie in an interval of $\pm 2\sigma$ around the regression line?

Multiple linear regression

- Linear regression with k explanatory variables:

$$Y = \alpha + \sum_i \beta_i X_i + \varepsilon, \tag{1}$$

- The coefficients $\beta_1, ..., \beta_k$ measure the *marginal effects* of each explanatory variable, i.e. the effect of explanatory variable while keeping fixed all the remanining ones.

Linear regression is not a causal model

Consider the model

$$\text{earnings} = -26.0 + 0.6 * \text{height} + \text{error}$$

To say that "the effect of height on earnings" is 600 is to suggest that, if we were to increase someone's height by one inch, his or her earnings would increase by an expected amount of $600. But the model has captured instead an an observational pattern, that taller people in the sample have higher earnings on average.

- Correct interpretation: the average difference in earnings, between two groups of people whose height differ by 1 inch, is 600 dollars.

Probabilistic regression

- The prediction account for the uncertainty of the estimated parameters (α, β, σ)
- Hierarchical regression: learning related regression models for different sources of data e.g., different hospitals applying the same treatment.
- Possibility of adopting a *robust regression* to deal with outliers.

Probabilistic regression

$$Y \sim N(\mu = \alpha + X\beta, \sigma)$$

- We must specify a prior distribution for each of parameter: α, β, σ.

Probabilistic regression

- We independently specify the prior of each parameter.
- Ideally, we define the priors based on background information
- If this is not possible, it is recommended setting *weakly informative* which define the order of magnitude of the parameters, after having scaled the data.

Using background information

Define a regression model for a bike sharing company, based on:

- For every one degree increase in temperature, rides increases by about 100 rides; the average increase is between 20 and 180.
- On an average temperature day (65 - 70 degrees), there are around 5000 riders, though this varies between 3000 and 7000.
- At any given temperature, daily ridership varies with a moderate standard deviation of 1250 rides.

Prior for the slope β

For every one degree increase in temperature, ridership typically increases by 100 rides; the average increase is between 20 and 180.

$$\beta \sim N(100, 40)$$

Prior for the intercept α

- We have no information about the intercept, i.e., the average number of rides when the temperature is 0).
- We know however the expected number of rides given an average temperature.
  On an average temperature day, there are around 5000 riders, though this could vary between 3000 and 7000.
- To use this information, we *center* the temperature X:

$$X_c = X - \bar{x}$$

where $\bar{x}$ is the average temperature in the sample.

Regression with centered data

$$
\begin{aligned}
Y &= \alpha + \beta X \\
Y &= \alpha + \beta \underbrace{(X - \bar{x})}_{X_c} + \beta \bar{x} \\
Y &= \alpha + \beta X_c + \beta \bar{x} \\
Y &= \underbrace{\alpha + \beta \bar{x}}_{\alpha_c} + \beta X_c \\
Y &= \alpha_c + \beta X_c
\end{aligned}
$$

- The intercept with centered data α_c is the expected value of Y when $X_c = 0$ i.e., when $X = \bar{x}$.
- α_c is the average value of Y when X is at its mean.

Centering yields also better sampling

- Centering the explanatory variables helps to numerically sample the posterior distribution of the parameters.
- Sampling the posterior of the regression model on the raw data might be slow and inefficient (low ESS, equivalent sample size).

Prior for α_c

On an average temperature day (65 - 70 degrees), there are around 5000 riders, though this could vary between 3000 and 7000.

- $\alpha_c \sim N(5000, 1000)$.

Prior for sigma

At any given temperature, daily ridership will tend to vary with a moderate standard deviation of 1250 rides.

Prior for sigma

```
from scipy.stats import halfnorm
pd.DataFrame(halfnorm.rvs(scale=1850, size=10000)).describe()
```

	0
count	10000.000000
mean	1485.546268
std	1118.971490
min	0.307154
25%	604.873887
50%	1258.085145
75%	2128.698084
max	7078.218157

The resulting model: bike rides as a function of temperature

$$
\begin{aligned}
\alpha_c &\sim N(5000, 1000) \\
\beta &\sim N(100, 40) \\
\sigma &\sim \text{HalfNormal}(1850) \\
Y &\sim N(\alpha_c + \beta X_c, \sigma)
\end{aligned}
$$

- To fit the model it is necessary to use the centered covariate $X_c = X - \bar{x}$, where $\bar{x}$ is the sample mean.

Loading the data

```
#500 rows of data
bike_data = pd.read_csv('data/bikes.csv')
rides = bike_data["rides"]
temperature = bike_data["temp_actual"]
bike_data.head()
```

	date	season	year	month	day_of_week	weekend	holiday	t
0	2011-01-01	winter	2011	Jan	Sat	True	no	
1	2011-01-03	winter	2011	Jan	Mon	False	no	
2	2011-01-04	winter	2011	Jan	Tue	False	no	
3	2011-01-05	winter	2011	Jan	Wed	False	no	
4	2011-01-07	winter	2011	Jan	Fri	False	no	

Linear regression in PyMC3

```
#centered covariate
temperature_c = temperature - temperature.mean()

with pm.Model() as reg_model:
    alpha_c  = pm.Normal ('alpha_c', mu=5000, sigma= 1000)
    beta     = pm.Normal ('beta', mu=100, sigma= 40)
    sigma    = pm.HalfNormal ('sigma', sigma=1850 )
    y_hat    = alpha_c + beta * temperature_c
    y_pred = pm.Normal('y_pred', mu=yhat, sigma=sigma, observed=
    trace = pm.sample(return_inferencedata=True)
```