# Computational Statistics and Data Analysis (MVComp2)

**Solutions to exercise 7**

**Lecturer** Tristan Bereau

**Semester** Wi23/24
**Due** Dec. 7, 2023, 23:59

## 1 Poisson regression (2 points)

Consider a response variable defined on the positive integer domain, $y_n \in \{0, 1, \dots\}$. We propose to fit a model using Poisson regression, such that the distribution's parameter $\lambda_n = \lambda_n(\boldsymbol{w}^\top \boldsymbol{x}_n)$ is a linear function of the input variables.

(a) Show that you can write Poisson regression as a generalized linear model (GLM).

(b) Use the GLM to determine the first two moments.

### 1.1 Solution

(a) The Poisson distribution is given by

$$p(y_n|\boldsymbol{x}_n, \boldsymbol{w}) = \mathrm{e}^{-\lambda_n} \frac{\lambda_n^{y_n}}{y_n!}$$

The log pdf is thus given by

$$\begin{aligned}\log p(y_n|\boldsymbol{x}_n, \boldsymbol{w}) &= y_n \log \lambda_n - \lambda_n - \log(y_n!) \\ &= y_n \eta_n - A(\eta_n) + h(y_n)\end{aligned}$$

where we associate $\eta_n = \log(\lambda_n) = \boldsymbol{w}^\top \boldsymbol{x}_n$ to ensure that the natural parameter is a linear function of the inputs. Thus we have $\lambda_n = \exp(\boldsymbol{w}^\top \boldsymbol{x}_n)$. Further we have $A(\eta_n) = \lambda_n = \mathrm{e}^{\eta_n}$, and $h(y_n) = -\log(y_n!)$.

(b) The first two moments are given by

$$\begin{aligned}E[y_n|\boldsymbol{x}_n, \boldsymbol{w}] &= \frac{\mathrm{d}A}{\mathrm{d}\eta_n} = \mathrm{e}^{\eta_n} = \lambda_n \\ \mathrm{Var}[y_n|\boldsymbol{x}_n, \boldsymbol{w}] &= \frac{\mathrm{d}^2 A}{\mathrm{d}\eta_n^2} = \mathrm{e}^{\eta_n} = \lambda_n\end{aligned}$$

# 2 Binary-output linear regression (3 points)

Suppose we have binary input data, $x_i \in \{0, 1\}$ and output two-dimensional response vector, $y_i \in \mathbb{R}^2$. The data is the following

| $x$ | $y$ |
|---|---|
| 0 | $(-1, -1)^\top$ |
| 0 | $(-1, -2)^\top$ |
| 0 | $(-2, -1)^\top$ |
| 1 | $(1, 1)^\top$ |
| 1 | $(1, 2)^\top$ |
| 1 | $(2, 1)^\top$ |

Embed each $x_i$ into two dimensions using the following basis function

$$\phi(0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \phi(1) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

The model becomes $\hat{y} = \boldsymbol{W}\phi(x)$, where $\boldsymbol{W}$ is a $2 \times 2$ matrix. Compute the maximum likelihood estimator for $\boldsymbol{W}$.

## 2.1 Solution

Recall the solution for an ordinary least squares problem

$$\boldsymbol{W} = (\Phi^\top \Phi)^{-1} \Phi^\top \boldsymbol{y},$$

where $\Phi$ uses the basis function $\phi(x)$ over the data

$$\Phi = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}.$$

The product yields

$$\Phi^\top \Phi = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}$$

and its inverse simply yields

$$(\Phi^\top \Phi)^{-1} = \begin{bmatrix} 1/3 & 0 \\ 0 & 1/3 \end{bmatrix}.$$

Moreoever, the other product is given by

$$\Phi^\top Y = \begin{bmatrix} -4 & -4 \\ 4 & 4 \end{bmatrix}$$

such that

$$\boldsymbol{W} = \begin{bmatrix} -4/3 & -4/3 \\ 4/3 & 4/3 \end{bmatrix}.$$

## 3 Posterior credible interval (3 points)

The Bayesian analog of a confidence interval is called a credible interval. Let's work with that here. Consider $X \sim \mathcal{N}(\mu, \sigma^2 = 4)$. The mean, $\mu$, is unknown, but has a prior $\mu \sim \mathcal{N}(\mu_0, \sigma_0^2 = 9)$. After seeing $n$ samples the posterior is $\mu \sim \mathcal{N}(\mu_n, \sigma_n^2)$.

(a) Determine $\mu_n$ and $\sigma_n^2$.

(b) How big does $n$ have to be to ensure

$$p(a \leq \mu_n \leq b|D) \geq 0.95,$$

where $(a, b)$ is an interval centered on $\mu_n$ of width 1 and $D$ is the data?

**Hint**: 95% of the probability mass of a Gaussian is within $\pm 1.96\sigma$ of the mean.

### 3.1 Solution

(a) Write the posterior for $n$ datapoints

$$\begin{aligned}
p(\mu|X) &\propto p(X|\mu)p(\mu) \\
&= \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(X_i - \mu)^2\right)\exp\left(-\frac{1}{2\sigma_0^2}(\mu - \mu_0)^2\right) \\
&\propto \exp\left(-\frac{1}{2}\left(\frac{n}{\sigma^2} + \frac{1}{\sigma_0^2}\right)\mu^2 + \left(\frac{n\bar{X}}{\sigma^2} + \frac{\mu_0}{\sigma_0^2}\right)\mu + ...\right)
\end{aligned}$$

where the ... indicate terms that do not involve $\mu$. From this expression we identify the parameters of a Gaussian, $\mathcal{N}(\mu_n, \sigma_n^2)$

$$\begin{aligned}
\sigma_n^2 &= \left(\frac{n}{\sigma^2} + \frac{1}{\sigma_0^2}\right)^{-1} \\
\mu_n &= \sigma_n^2\left(\frac{n\bar{X}}{\sigma^2} + \frac{\mu_0}{\sigma_0^2}\right).
\end{aligned}$$

(b) The width of the interval $(a, b)$ is 1, centered around the mean. The half-width is thus 0.5, which corresponds to $1.96\sigma_n$. Therefore

$$\begin{aligned}
1.96\sigma_n &= 0.5 \\
\sigma_n &= 0.2551
\end{aligned}$$

Now solve for $n$

$$\begin{aligned}
\sigma_n^2 &= \left(\frac{n}{\sigma^2} + \frac{1}{\sigma_0^2}\right)^{-1} \\
n &= \left(\frac{1}{\sigma_n^2} - \frac{1}{\sigma_0^2}\right)\sigma^2
\end{aligned}$$

Plug in the provided values to yield $n \approx 61.022$. Therefore, $n$ must be at least 62 to ensure the condition on the credible interval.

# 4 Integration by Monte Carlo (2 points)

Estimate

$$\ell = \int_0^1 \mathrm{d}x \int_0^1 \mathrm{d}y \, \frac{\sin(x)\mathrm{e}^{-(x+y)}}{\ln(1+x)}$$

via Monte Carlo, and give a 95% confidence interval.

## 4.1 Solution

```
import numpy as np

# Define the function to integrate
def integrand(x, y):
    return np.sin(x) * np.exp(-(x + y)) / np.log(1 + x)

# Number of samples for Monte Carlo
n_samples = 1000000

# Generate random samples for x and y
x_samples = np.random.uniform(0, 1, n_samples)
y_samples = np.random.uniform(0, 1, n_samples)

integrand_values = integrand(x_samples, y_samples)
integral_estimate = np.mean(integrand_values)
integrand_std = np.std(integrand_values)
standard_error = integrand_std / np.sqrt(n_samples)

# 95% confidence interval for the mean estimate
confidence_interval = (
  integral_estimate - 1.96 * standard_error,
  integral_estimate + 1.96 * standard_error
)

print(
  f"estimate {integral_estimate:.4f}, " \
  f"interval: ({confidence_interval[0]:.4f}, {confidence_interval[1]:.4f})"
)
```

```
estimate 0.4549, interval: (0.4546, 0.4553)
```