2021F_Week05

September 22, 2021

0.1 Week05

Lets continue discussing the detials of Regression. In particular, we can write down the maximum likelhood estimators for β_0, β_1, and σ^2. Next we will write down the Fisher Infromation **Matrix** for a 2X1 vector that contains β_0 and β_1.

With our MLE and Fisher Infromation Matrix we can compute confidence intervals for β and setup hypothesis tests.

Our goal this week is to: 1. Write down the solution to the loglikelihood 2. Write down and understand how to use the Fisher Information 3. Learn how to build Confidnece intervals for β 4. Learn how to compute fitted values $\mu(\hat{x})$ 5. Briefly discuss the "true" model

0.1.1 The solution to the loglikelihood (ie the MLEs for β_0, β_1, and σ^2)

The MLE of β_1 ($\hat{\beta}_1$) is

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{N}(x_i - \bar{x})^2} \tag{1}$$

The MLE of β_0 ($\hat{\beta}_0$) is

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1\bar{x} \tag{2}$$

The MLE for σ^2 is worth some discussion of the model form for SLR. Recall SLR model form is

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i \tag{3}$$
$$\epsilon_i \sim \mathcal{N}\left(0, \sigma^2\right) \tag{4}$$

In model form we see that given estimates for β_0 and β_1 we can compute ϵ_i values. Lets plug in our MLE estimates for β_0 and β_1 and solve for ϵ_i:

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i \tag{5}$$
$$y_i = \hat{\beta}_0 + \hat{\beta}_1 x_i + \epsilon_i \tag{6}$$
$$\epsilon_i = y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 x_i\right) \tag{7}$$

We can consider each $\epsilon\_\{i\}asadatapoint.Ourdataise\epsilon = [\epsilon_1, \epsilon_2, \cdots, \epsilon_N]$ and our model is

$$E_i \sim \mathcal{N}\left(0, \sigma^2\right) \tag{8}$$

We know from previous work that the MLE for σ^2 of this model is

$$\hat{\sigma}^2 = \frac{(\epsilon_i - \bar{\epsilon})^2}{N} \tag{9}$$

and by assumption, and the Law of Large Numbers, we expect $\bar{\epsilon} \approx 0$ sp

$$\hat{\sigma}^2 = \frac{(\epsilon_i - 0)^2}{N} \tag{10}$$

$$\hat{\sigma}^2 = \frac{\epsilon_i^2}{N} \tag{11}$$

$$\tag{12}$$

Though the above is the MLE for ^{2} it is more traditional to estimate $\hat{\sigma}^2$ as

$$\hat{\sigma}^2 = \frac{\epsilon_i^2}{N-2} \tag{13}$$

$$\tag{14}$$

0.1.2 Fitted values

Assume data $\mathcal{D} = [(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)]$ and the following model

$$Y_i | x_i \sim \mathcal{N}\left(\mu(x_i), \sigma^2\right) \tag{15}$$

where

$$\mu(x_i) = \beta_0 + \beta_1 x_i \tag{16}$$

Then $E(Y_i|x_i) = \mu(x_i) = \beta_0 + \beta_1 x_i$ and the MLE estimate of $\mu(x_i)$, denoted $\hat{\mu}(x_i)$ is called a **fitted value** for Y_i. A **fitted value** for Y_i assuming a SLR model is

$$\hat{\mu}(x_i) = \hat{\beta}_0 + \hat{\beta}_1 x_i \tag{17}$$

0.1.3 Treating parameters as a vector and the Fisher Information Matrix

Up until now we have specificed our data and model for individual data points/random variables. Let us reformulate our data and model using Matrix algebra (because it is awesome).

Assume data $\mathcal{D} = [(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)]$ Our model says that each random variable Y_i has the following distribution

$$Y_1 = 1\beta_0 + x_1\beta_1 + \epsilon_1 \tag{18}$$
$$Y_2 = 1\beta_0 + x_2\beta_1 + \epsilon_2 \tag{19}$$
$$Y_3 = 1\beta_0 + x_3\beta_1 + \epsilon_3 \tag{20}$$
$$Y_4 = 1\beta_0 + x_4\beta_1 + \epsilon_4 \tag{21}$$
$$Y_5 = 1\beta_0 + x_5\beta_1 + \epsilon_5 \tag{22}$$

where each $\epsilon_i \sim \mathcal{N}\left(0, \sigma^2\right)$.

This model can be rewritten as

$$Y = X\beta + \epsilon \tag{23}$$

where

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \\ Y_5 \end{bmatrix} \tag{24}$$

$$\epsilon = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \end{bmatrix} \tag{25}$$

where each ϵ_i is normally distributed with parameter σ^2 or $\epsilon_i \sim \mathcal{N}\left(0, \sigma^2\right)$.

$$X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ 1 & x_4 \\ 1 & x_5 \end{bmatrix} \tag{26}$$

and finally

$$\beta = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} \tag{27}$$

0.1.4 The Multivariate Normal distribution and probabilities over vectors

Because we have re-expressed our Simple linear regression using matrices and vectors, we need a way to talk about probabilities over vectors. To be clear, we rewrote our SLR

$$Y \sim \text{Some Distribution}(X\beta, \Sigma) \tag{28}$$

But how should we characterize a vector where each entry in the vector has a Normal Distribution?

The Multivariate Normal Distribution (MVN) The multivariate distribution uses a probability density over vectors to assign with assigning probabilities over intervals. For a **random vector** $x = [x_1, x_2, \cdots, x_N]'$ the MVN Normal distribution is denoted

$$x \sim MVN\left(\mu, \Sigma\right) \tag{29}$$

where $\mu = [\mu_1, \mu_2, \cdots, \mu_N]$ is a mean vector and

$$\Sigma = \begin{bmatrix} \sigma^2_{x_1} & \sigma_{x_1,x_2} & \cdots & \sigma_{x_1,x_N} \\ \sigma_{x_1,x_2} & \sigma^2_{x_2} & \cdots & \sigma_{x_2,x_N} \\ \vdots & & & \end{bmatrix} \tag{30}$$

is called the covariance matrix where agonal entries in this matrix Σ_{ii} are equal to the variance of x_i and entries Σ_{ij} are equal to the covariance between x_i and x_j.

PDF The probability density for the multivariate normal distribution over vectors of length k is

$$f(x, \mu, \Sigma) = (2\pi)^{-k/2} |\Sigma|^{-1/2} \exp -\frac{1}{2} \left[(x-\mu)'\Sigma^{-1}(x-\mu) \right] \tag{31}$$

The Expected value and Coviariance For a random vector X of length k with a $MVN(\mu, \Sigma)$ distribution the expected value is

$$E(X) = \mu \tag{32}$$

where $\mu = [\mu_1, \mu_2, \cdots, \mu_k]'$ is a vector of length k.

The covariance matrix (rather than variance) of a random variable X as above is

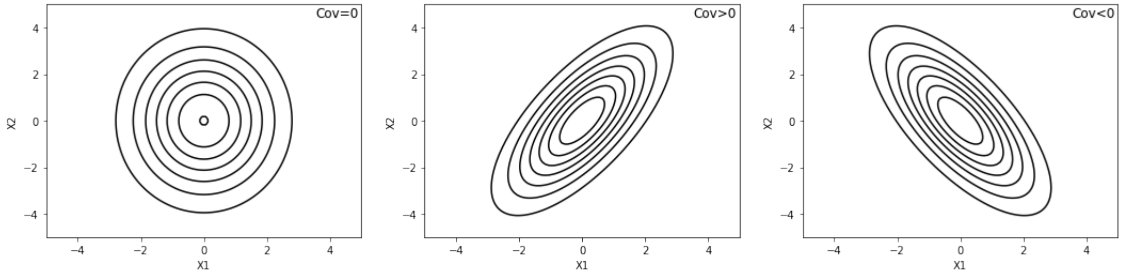
$$Cov(X) = \Sigma \tag{33}$$

For a vector of length two we can visualize the probability density function of the MVN for different parameter values.

[42]:

```
import scipy.stats

x = np.linspace(-5, 5, 100)
y = np.linspace(-5, 5, 100)

X, Y = np.meshgrid(x, y)

def MVN(pos,m,s):
    return scipy.stats.multivariate_normal(mean=m,cov=s).pdf(pos)

pos = np.empty(X.shape + (2,))
pos[:, :, 0] = X
pos[:, :, 1] = Y

Z= MVN(pos,m=[0,0],s=[[2,0],[0,4]])

fig,axs = plt.subplots(1,3)

ax = axs[0]
ax.contour(X, Y, Z, colors='black');

ax.text(0.99,0.99,r"Cov=0",transform=ax.
 ↪transAxes,ha="right",va="top",fontsize=12)
ax.set_xlabel("X1")
ax.set_ylabel("X2")


Z= MVN(pos,m=[0,0],s=[[2,2],[2,4]])

ax = axs[1]
ax.contour(X, Y, Z, colors='black');

ax.text(0.99,0.99,r"Cov>0",transform=ax.
 ↪transAxes,ha="right",va="top",fontsize=12)
ax.set_xlabel("X1")
ax.set_ylabel("X2")

Z= MVN(pos,m=[0,0],s=[[2,-2],[-2,4]])
```

```
ax = axs[2]
ax.contour(X, Y, Z, colors='black');

ax.text(0.99,0.99,r"Cov<0",transform=ax.
 ↪transAxes,ha="right",va="top",fontsize=12)
ax.set_xlabel("X1")
ax.set_ylabel("X2")

fig.set_size_inches(18,4)
```



On the far left is a MVN density with 0 covariance between X1 and X2, where

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \sim MVN\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix}\right) \tag{34}$$

The middle figure is a MVN distribution with a positive covariance

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \sim MVN\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 2 & 2 \\ 2 & 4 \end{bmatrix}\right) \tag{35}$$

and the figure on the far right is MVN with a negative covariance

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \sim MVN\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 2 & -2 \\ -2 & 4 \end{bmatrix}\right) \tag{36}$$

0.1.5 Applying the MVN distributoin to our linear regression

Recall that our linear regression model can be written

$$Y = X\beta + \epsilon \tag{37}$$

where X is a matrix and Y, β, and ϵ are vectors.

Distribution for MLE of β The distribution over $\hat{\beta}$ is

$$\beta = \begin{bmatrix}\beta_0 \\ \beta_1\end{bmatrix} \sim MVN\left(\hat{\beta}, \hat{\sigma}^2 \left(X'X\right)^{-1}\right) \tag{38}$$

where $\hat{\beta}$, the MLE for β_0 and for β_1 can be written using matrices

$$= (X'X)^{-1}X'y \tag{39}$$

Because $X\hat{\beta}$ is the μ we can write our fitted values $\hat{y}$ as

$$\begin{aligned}\hat{y} &= X\left[(X'X)^{-1}X'y\right] && (40)\\ &= X\left[(X'X)^{-1}X'\right]y && (41)\\ &= P_x y && (42)\\ & && (43)\end{aligned}$$