PENDUGAAN DATA HILANG DENGAN MENGGUNAKAN *DATA AUGMENTATION*

Mesra Nova[1], Moch. Abdul Mukid[2]
[1]Alumni Program Studi Statistika UNDIP
[2]Staf Pengajar Program Studi Statistika UNDIP
mamukid@yahoo.com

Abstract

Data augmentation is a method for estimating missing data. It is a special case of Gibbs sampling which has two important steps. The first step is imputation or I-step where the missing data is generated based on the conditional distributions for missing data if the observed data are known. The next step is posterior or P-step where the estimation process of parameter values from the complete data is conducted. Imputation and posterior steps on the data augmentation will continue to run until the convergence is reached. The estimate of missing data is obtained through the average of simulated values.

Keywords: Missing Data, Data Augmentation, Imputation Step, Posterior Step

1. Pendahuluan

Data hilang adalah informasi yang tidak tersedia untuk sebuah kasus. Data hilang dapat terjadi karena informasi yang dibutuhkan untuk sesuatu pada satu atau beberapa variabel tidak diberikan, sulit dicari atau memang informasi tersebut tidak tersedia. Beberapa alasan mengapa data tersebut dapat hilang diantaranya adalah mungkin hilang karena peralatan tidak berfungsi, cuaca buruk, orang yang diamati sakit, atau data tidak dimasukkan dengan benar[4].

Jika data yang diperoleh dalam suatu penelitian tidak mengandung data hilang maka parameter populasi akan mudah diduga dengan metode maksimum likelihood. Jika data yang diperoleh mengandung data hilang maka fungsi likelihood dari data yang teramati akan menjadi sulit untuk dimaksimumkan. Fungsi likelihood adalah fungsi densitas bersama dari n sampel random $Y_1, Y_2, ..., Y_n$ dan dinyatakan dalam bentuk $f(y_1, y_2, ..., y_n \mid \theta)$. Jika data lengkap yang diperoleh adalah berdistribusi normal maka penduga maksimum likelihood untuk mean dari data tersebut adalah penjumlahan dari nilai data dalam satu variabel dibagi dengan ukuran data yaitu $\bar{y}$[1]. Namun ketika ada beberapa nilai dari variabel tersebut yang hilang maka penjumlahan dari nilai data variabel tersebut tidak diketahui, dan penghitungan terhadap dugaan untuk mean dari variabel tersebut tidak dapat dilakukan. Penggunaan metode maksimum likelihood untuk kasus yang mengandung data hilang akan menyebabkan hasil estimasi parameter menjadi bias dan tidak efisien[5].

Pendekatan yang paling sederhana untuk mengatasi data hilang yaitu dengan metode *listwise deletion* dan *pairwise deletion*. Kedua metode ini menyarankan untuk menghapus nilai data yang hilang dan tidak melibatkannya dalam analisis selanjutnya. Ketika kasus nilai hilang pada data hanya terdiri dari sebagian kecil dari semua data maka metode penghapusan yaitu *listwise deletion* dan *pairwise deletion* dapat menjadi solusi yang masuk akal. Namun, jika nilai data yang hilang terjadi pada sebagian besar data maka metode penghapusan akan menyebabkan sejumlah informasi dari data akan terbuang

begitu saja[6]. Oleh karena itu pendugaan terhadap nilai data hilang menjadi alternatif pilihan yang layak untuk dilakukan.

Pendekatan yang biasa digunakan untuk menduga nilai data hilang diantaranya adalah *mean imputation*, *hot deck imputation*, c*old decki imputation*, *regression imputation*, s*ubstitution*, dan c*omposite methods*[4]. Untuk pendekatan yang lebih modern terdapat metode untuk mengatasi data hilang diantaranya adalah dengan menggunakan algoritma *Expectation Maximization* (EM) atau dengan menggunakan metode *Markov Chain Monte Carlo* (MCMC) yang terdiri atas algoritma *Data Augmentation* (DA), *Gibbs Sampling*, *Metropolis-Hasting*, dan algoritma terkait lainnya[6]. Algoritma DA yang merupakan salah satu bagian dari metode MCMC adalah algoritma yang akan digunakan dalam penyelesaian masalah data hilang pada penulisan makalah ini.

2. Tinjauan Pustaka

Pada bagian ini akan dibahas mengenai berbagai konsep yang berkaitan dengan *Data Augmentation* (DA), yaitu mengenai prosedur DA, penerapan DA untuk estimasi data hilang, konvergensi DA dan penerapan DA untuk data berdistribusi normal multivariate.

2.1 Prosedur Data Augmentation

Data augmentation (DA) adalah sebuah algoritma untuk membangkitkan data dari distribusi tertentu. DA dipandang sebagai kasus khusus dari Gibbs sampling yang terdiri atas dua langkah. Dua langkah Gibbs sampling tersebut merupakan proses penarikan sampel berdasarkan dua distribusi bersyarat. Metode ini diperkenalkan oleh Tanner dan Wong yang menetapkan proses berulang untuk mendapatkan perkiraan suatu parameter tertentu berdasarkan distribusi posteriornya[7]. Berikut ini akan dijelaskan tentang prosedur DA.

Misalkan vektor acak Z dipartisi menjadi dua subvektor yaitu $\mathrm{z} = (\mathrm{u}, \mathrm{v})$, dimana distribusi bersama $\mathrm{P}(\mathrm{z})$ akan lebih mudah disimulasikan jika dipartisi menjadi distribusi bersyarat $\mathrm{P}(\mathrm{u}|\mathrm{v}) = \mathrm{g}(\mathrm{u}|\mathrm{v})$ dan $\mathrm{P}(\mathrm{v}|\mathrm{u}) = \mathrm{h}(\mathrm{v}|\mathrm{u})$. Pada iterasi ke-$t$ diberikan vektor acak Z sebagai berikut

$$\mathrm{Z}^{(t)} = \left(\mathrm{z}_1^{(t)}, \mathrm{z}_2^{(t)}, \dots, \mathrm{z}_\mathrm{m}^{(t)}\right) \tag{1}$$

yang merupakan sampel berukuran m dari distribusi yang mendekati distribusi target $\mathrm{P}(\mathrm{z})$, yang kemudian dipartisi menjadi

$$\left(\left(\mathrm{u}_1^{(t)}, \mathrm{v}_1^{(t)}\right), \left(\mathrm{u}_2^{(t)}, \mathrm{v}_2^{(t)}\right), \cdots, \left(\mathrm{u}_\mathrm{m}^{(t)}, \mathrm{v}_\mathrm{m}^{(t)}\right)\right) \tag{2}$$

Dengan menggunakan DA, sampel di atas akan diperbaharui melalui dua langkah berikut:

1. Langkah pertama, pada iterasi ke $t+1$, sampel acak dari U diambil dari distribusi bersyarat $\mathrm{g}\left(\mathrm{u}|\mathrm{v}_\mathrm{i}^{(t)}\right)$ dan menghasilkan

   $$\mathrm{U}^{(t+1)} = \left(\mathrm{u}_1^{(t+1)}, \mathrm{u}_2^{(t+1)}, \dots, \mathrm{u}_\mathrm{m}^{(t+1)}\right) \tag{3}$$

   yang saling bebas untuk $\mathrm{i} = 1,2, \cdots, \mathrm{m}$.

2. Langkah kedua, pada iterasi ke $t+1$, yaitu sampel acak dari V diambil dari distribusi bersyarat $\mathrm{h}\left(\mathrm{v}|, \mathrm{u}_\mathrm{i}^{(t+1)}\right)$,

   $$\mathrm{V}^{(t+1)} = \left(\mathrm{v}_1^{(t+1)}, \mathrm{v}_2^{(t+1)}, \dots, \mathrm{v}_\mathrm{m}^{(t+1)}\right) \tag{4}$$

Hasil dari dua langkah pada DA di atas akan melengkapi sampel baru dengan bentuk sebagai berikut:

$$Z^{(t+1)} = \left(\left(u_1^{(t+1)}, v_1^{(t+1)}\right), \cdots, \left(u_m^{(t+1)}, v_m^{(t+1)}\right)\right) \quad (5)$$

Dengan menggunakan analisis fungsional, Tanner dan Wong menunjukkan bahwa distribusi dari $Z^{(t)}$ konvergen kepada $P(z)$ ketika $t \to \infty$[7].

2.2 Penerapan *Data Augmentation* untuk Pendugaan Data Hilang

Data Augmentation (DA) merupakan algoritma yang diterapkan untuk menduga nilai data hilang melalui pendekatan Bayesian. DA menganggap bahwa mekanisme hilangnya data adalah *Missing at Random*, yaitu peluang data yang hilang tidak tergantung pada nilai data yang hilang, tetapi tergantung pada nilai data yang teramati.

Jika Y_{obs} adalah seluruh nilai data yang teramati dan Y_{mis} adalah notasi untuk nilai data yang hilang serta θ adalah parameter pembangkit data maka distribusi *posterior* pada masalah data hilang $P(\theta|Y_{obs})$ biasanya akan sulit untuk disimulasikan secara langsung. Tetapi jika Y_{obs} kemudian ditambah oleh nilai yang diasumsikan dari Y_{mis}, maka posterior $P(\theta|Y_{obs}, Y_{mis})$ akan menjadi lebih mudah untuk disimulasikan. Berikut ini akan dijelaskan skema sampling iteratif pada DA.

Jika pada iterasi ke-t diberikan *parameter* $\theta^{(t)}$, maka selanjutnya dapat dibangkitkan sebuah dugaan dari nilai data yang hilang dari distribusi prediksi bersyarat Y_{mis} yaitu:

$$Y_{mis}^{(t)} \sim P\left(Y_{mis}|Y_{obs}, \theta^{(t)}\right) \quad (6)$$

Kemudian dengan memperhatikan $Y_{mis}^{(t)}$, nilai baru untuk θ pada iterasi ke $(t+1)$ diambil dari distribusi posterior jika data lengkap diketahui yaitu:

$$\theta^{(t+1)} \sim P\left(\theta|Y_{obs}, Y_{mis}^{(t)}\right) \quad (7)$$

Ulangi langkah (6) dan (7) dengan sebuah nilai awal $\theta^{(0)}$. Hasil akhirnya akan membentuk sebuah barisan stochastic $\left\{\left(\theta^{(t)}, Y_{mis}^{(t)}\right): t = 1,2,\cdots\right\}$ yang mempunyai distribusi $P(\theta, Y_{mis}|Y_{obs})$. Sementara untuk sub rangkaian $\left\{\theta^{(t)}: t = 1,2,\cdots\right\}$ dan $\left\{Y_{mis}^{(t)}: t = 1,2,\cdots\right\}$ masing-masing akan mempunyai distribusi $P(\theta|Y_{obs})$ dan $P(Y_{mis}|Y_{obs})$.

Tahapan yang bersesuaian pada persamaan (6) disebut sebagai tahap Imputasi atau step-I dan persamaan (7) sebagai tahap Posterior atau step-P. Persamaan (6) menunjukkan proses imputasi mencari nilai data hilang Y_{mis} dan persamaan (7) sesuai untuk menggambarkan proses mencari nilai θ yang baru dari posterior jika data lengkap diketahui. Tahap-Imputasi dan tahap Posterior akan terus berjalan sampai didapatkan kekonvergenan untuk θ.

2.3 Konvergensi *Data Augmentation*

Data Augmentation (DA) akan menghasilkan rangkaian nilai-nilai simulasi $(\theta^{(t)}, Y_{mis}^{(t)})$ yang cukup panjang. Nilai-nilai simulasi tersebut merupakan nilai yang diduga dari distribusi posterior, dan akan konvergen kepada distribusi yang dituju yaitu distribusi $p(Y_{mis}, \theta|Y_{obs})$[6].

Iterasi yang dilakukan dengan mengulangi langkah pada persamaan (6) dan (7) pada periode T tertentu akan menghasilkan nilai-nilai yang tidak dipengaruhi oleh nilai awal dan distribusi dari iterasi yang baru tersebut akan mendekati distribusi posteriornya. Tahap awal ini disebut dengan proses *burn-in period*, yang berguna untuk melepaskan ketergantungan pada nilai awal dan distribusi awal. Salah satu cara untuk memperkirakan panjang *burn-in period* adalah dengan memeriksa plot *time-series* dari simulasi rata-rata untuk setiap variabel terhadap jumlah iterasinya[3].

Plot *time-series* merupakan plot simulasi nilai rata-rata setiap variabel yang berturut-turut terhadap jumlah iterasinya. Adanya trend jangka panjang dalam plot menunjukkan bahwa iterasi yang berurutan sangat berkorelasi dan bahwa serangkaian iterasinya belum mencapai konvergensi. Kecenderungan peningkatan atau penurunan nilai rata-rata dalam plot *time-series* menunjukkan bahwa *burn-in period* belum berakhir. Tidak adanya trend naik atau trend turun pada plot merupakan keadaan yang ideal yang menunjukkan bahwa rata-rata setiap variabel setelah proses iterasi pada DA tersebut konvergen kepada distribusi posterior yang dituju[3].

Penilaian konvergensi DA dapat juga dilakukan dengan mengukur tingkat ketergantungan antara rantai simulasi yang berturut-turut atau nilai autokorelasinya. Autokorelasi mengkuantifikasi besarnya ketergantungan antar mean yang dihasilkan pada setiap iterasi. Autokorelasi mengukur korelasi antara kumpulan nilai simulasi $\left\{Y_j^{(t)}\right\}$ dan $\left\{Y_j^{(t+L)}\right\}$, dimana L adalah lag yang memisahkan kumpulan dua nilai dan k adalah ukuran sampel yang digunakan. Fungsi autokorelasi untuk variabel tertentu dapat dihitung sebagai fungsi dari nilai-nilai lag L yang berbeda. Untuk variabel j, autokorelasi lag L dapat dihitung dengan:

$$r_{jL} = \frac{k}{k-L}\frac{\sum_{j=1}^{k-L}(Y_j-\bar{Y})(Y_{j+L}-\bar{Y})}{\sum_{j=1}^{m}(Y_i-\bar{Y})^2} \tag{8}$$

dengan $\bar{Y}$ adalah rata-rata variabel setelah Y_{mis} disubstitusi dengan nilai imputasi[3].

Fungsi autokorelasi dapat ditampilkan dalam bentuk grafis yaitu plot autokorelasi. Plot autokorelasi merupakan grafis yang menampilkan nilai-nilai autokorelasi pada sumbu vertikal dan nilai-nilai lag pada sumbu horizontal. Grafik dari hasil pemetaan tersebut dinamakan korelogram, yang dapat digunakan untuk memeriksa berautokorelasi atau tidaknya data deret waktu. Jika korelogram berpola acak, maka dapat disimpulkan bahwa data deret waktu tidak berautokorelasi[6]. Jika data deret waktu hasil iterasi dari proses DA tidak berautokorelasi, maka rata-rata dari setiap hasil iterasi akan bergerak stabil mengarah kepada distribusi yang dituju.

2.4 Inferensi pada *Data Augmentation*

Inferensi pada DA difokuskan pada pendugaan data yang hilang dengan menggunakan *Multiple Imputation* (MI). MI merupakan salah satu metode imputasi yang menghasilkan inferensi yang valid untuk nilai hilang $Y_{i(mis)}$. Setiap nilai $Y_{i(mis)}$ akan diisi oleh beberapa nilai (dua atau lebih) yang dibangkitkan melalui sejumlah imputasi. Banyaknya imputasi yang dilakukan pada proses MI disimbolkan dengan m. Sejumlah m nilai yang mengisi nilai hilang $Y_{i(mis)}$ akan membentuk m buah kelompok data yang telah lengkap.

Nilai dugaan akhir untuk $Y_{i(mis)}$ diperoleh dari rata-rata nilai simulasi $Y_{i(mis)}$ yang dipilih pada setiap m imputasi. Misalkan dalam satu imputasi dibangkitkan 5 imputasi

yang masing-masing terdiri atas 100 iterasi. Nilai $Y_{i,(mis)}$ yang dipilih adalah nilai yang berada pada urutan terakhir dari tiap satu imputasi. Jika dijalankan 5 imputasi yang masing-masing terdiri atas 100 iterasi, maka akan diperoleh 5 nilai dugaan $Y_{i,(mis)}$. Sehingga nilai dugaan $Y_{i,(mis)}$ yang digunakan untuk mengisi kasus data hilang $Y_{i,(mis)}$ adalah nilai rata-rata dari lima nilai $Y_{i,(mis)}$ tersebut.

2.5 Penerapan *Data* Augmentation pada Data Berdistribusi Normal Multivariat

Pada data yang berasal dari distribusi normal multivariat, DA diterapkan dengan pendekatan Bayesian dengan melakukan langkah-langkah sebagai berikut:

1. Langkah Imputasi (Step-I)

Pada step-I, nilai hilang akan disimulasikan untuk setiap pengamatan secara independen dengan memberikan sebuah estimasi vektor mean dan matrik kovarian. Jika variabel nilai hilang pada pengamatan ke-i adalah $Y_{i(mis)}$ dan variabel nilai yang teramati adalah $Y_{i(obs)}$, maka pada langkah-I akan ditarik nilai-nilai untuk $Y_{i(mis)}$ dari distribusi bersyarat untuk $Y_{i(mis)}$ jika $Y_{i(obs)}$ diketahui.

Pada iterasi pertama $t = 1$ untuk setiap pengamatan yang nilai datanya hilang, akan diberikan estimasi vektor mean dan matriks kovarian sebagai nilai parameter awal. Parameter awal untuk vektor mean dan matriks kovarian masing-masing akan didefinisikan dengan $\boldsymbol{\mu}^{(0)}$ dan $\boldsymbol{\Sigma}^{(0)}$. Nilai parameter awal ini akan diperoleh dengan langkah berikut ini.

Misalkan $\boldsymbol{\mu} = [\boldsymbol{\mu_1}, \boldsymbol{\mu_2}]'$ adalah vektor mean yang dipartisi untuk dua variabel yaitu Y_{obs} dan Y_{mis}, dimana μ_1 adalah vektor mean untuk variabel Y_{obs} dan μ_2 adalah vektor mean untuk variabel Y_{mis}. Juga dimisalkan bahwa matriks kovarian dipartisi menjadi

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma_{11}} & \boldsymbol{\Sigma_{12}} \\ \boldsymbol{\Sigma_{12}'} & \boldsymbol{\Sigma_{22}} \end{bmatrix} \tag{9}$$

dengan $\boldsymbol{\Sigma_{11}}$ adalah matriks kovarian untuk variabel Y_{obs}, $\boldsymbol{\Sigma_{22}}$ adalah matriks kovarians untuk variabel Y_{mis} dan $\boldsymbol{\Sigma_{12}}$ adalah matriks kovarian diantara variabel Y_{obs} dan variabel Y_{mis}.

Distribusi bersyarat dari $Y_{mis} = y_2$ jika diberikan $Y_{obs} = y_1$ adalah distribusi normal multivariat normal dengan vektor mean $\boldsymbol{\mu}^{(0)} = \boldsymbol{\mu_2} + \boldsymbol{\Sigma_{21}\Sigma_{11}^{-1}(y_1 - \mu_1)}$ dan matriks kovarian $\boldsymbol{\Sigma}^{(0)} = \boldsymbol{\Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}}$ (kovarian tidak tergantung pada harga $Y_{obs} = y_1$) dapat dituliskan sebagai berikut:

$$Y_2|Y_1 = y_1 \sim \mathbf{N}_q\left(\boldsymbol{\mu}^{(0)}, \boldsymbol{\Sigma}^{(0)}\right) \tag{10}$$

dapat juga dituliskan dengan:

$$Y_2|Y_1 = y_1 \sim \mathbf{N}_q\left(\boldsymbol{\mu_2} + \boldsymbol{\Sigma_{21}\Sigma_{11}^{-1}(y_1 - \mu_1)}, \boldsymbol{\Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}}\right) \tag{11}$$

Distribusi bersyarat dari $Y_{mis} = y_2$ jika diberikan $Y_{obs} = y_1$ diatas digunakan untuk membangkitkan nilai-nilai untuk data hilang pada step-I.

2. Langkah Posterior (Step-P)

Setelah data hilang pada step-I diduga, sehingga data menjadi lengkap maka akan dilanjutkan pada step-P . Step-P merupakan proses penarikan nilai posterior dari vektor mean dan matriks kovarian. Proses step-P dimulai dengan mengestimasi nilai vektor mean dan matriks kovarian menggunakan data lengkap yang dihasilkan pada step-I yang sebelumnya. Tujuan utama dari step-P adalah untuk estimasi sampel yang baru dari masing-masing distribusi posterior vektor mean dan matriks kovarian, sehingga pada step-I berikutnya dapat digunakan untuk memperbaharui nilai parameter.

Distribusi posterior untuk vektor mean dan matrik kovarian, pada kasus ini diperoleh dengan menggunakan prior non-informatif. Prior non-informatif untuk data berdistribusi normal multivariat dengan menggunakan metode Jeffry's menghasilkan $|\boldsymbol{\Sigma}|^{-\frac{1}{2}(p+1)}$. Distribusi posterior dapat ditentukan dengan rumus berikut:

$$f(\boldsymbol{\theta}|\mathbf{y}) = \frac{L(\boldsymbol{\theta})\times f(\boldsymbol{\theta})}{\int_{-\infty}^{\infty} L(\boldsymbol{\theta})\times f(\boldsymbol{\theta}) d\boldsymbol{\theta}} \tag{12}$$

Algoritma data augmentation menggunakan teknik simulasi Monte Carlo untuk menarik sebuah nilai yang baru untuk vektor mean $\boldsymbol{\mu}$ dan matriks $\boldsymbol{\Sigma}$ dari distribusi posterior**.**

Nilai posterior parameter $\boldsymbol{\Sigma}$ pada iterasi pertama diperoleh dengan cara menarik sebuah nilai posterior $\boldsymbol{\Sigma}^{(1)}$ dari distribusi bersyarat $\boldsymbol{\Sigma}^{(1)}$ jika nilai data lengkap yaitu Y_{obs} dan $Y_{mis}^{(1)}$ diketahui. Distribusi bersyaratnya adalah invers wishart dan dapat dituliskan sebagai berikut:

$$\boldsymbol{\Sigma}^{(1)}|Y_{obs}, Y_{mis}^{(1)} \sim W^{-1}(n-1, (n-1)S)$$

Simulasi monte carlo akan digunakan untuk menarik sebuah nilai matriks kovarian yang baru dari distribusi posterior untuk $\boldsymbol{\Sigma}^{(1)}$. Nilai matriks kovarian yang dihasilkan dari distribusi posterior $\boldsymbol{\Sigma}^{(1)}$, diperoleh dengan proses komputasi yaitu membangkitkan bilangan acak untuk distribusi *Inverse Wishart*. Distribusi *Inverse Wishart* tersebut dibangkitkan dengan derajat bebas $n-1$ yaitu jumlah unit pengamatan dikurangi satu dan dengan $(n-1)S$ adalah matriks *corrected sum of squares and cross products* (CSSCP) yaitu $\sum_{i=1}^{n}(y_i-\bar{y}_i)(y_i-\bar{y}_i)'$.

Algoritma DA menggunakan teknik yang sama untuk menghasilkan vektor mean $\boldsymbol{\mu}$ yang baru. Nilai posterior untuk vektor mean, dapat disimulasi dengan cara menarik sebuah nilai posterior $\boldsymbol{\mu}^{(1)}$ dari distribusi bersyarat $\boldsymbol{\mu}^{(1)}$ jika nilai Y_{obs}, $Y_{mis}^{(1)}$ dan nilai $\boldsymbol{\Sigma}^{(1)}$ diketahui. Distribusi bersyaratnya adalah distribusi normal dan dapat dituliskan sebagai berikut:

$$\boldsymbol{\mu}^{(1)}|(Y_{obs}, Y_{mis}^{(1)}, \boldsymbol{\Sigma}^{(1)}) \sim N\left(\bar{y}, \frac{1}{n}\boldsymbol{\Sigma}^{(1)}\right)$$

Simulasi monte carlo akan digunakan untuk menarik sebuah nilai vektor mean yang baru dari distribusi posterior untuk $\boldsymbol{\mu}^{(1)}$. Sehingga dari iterasi pertama $t=1$ diperoleh $\left(Y_{mis}^{(1)}, \left(\boldsymbol{\Sigma}^{(1)}, \boldsymbol{\mu}^{(1)}\right)\right)$.

Proses di atas merupakan proses DA yang pertama, setelah itu akan kembali kepada step-I berikutnya. Pada iterasi $t=2$, parameter yang dihasilkan dari proses simulasi dari nilai posterior vektor mean $\boldsymbol{\mu}^{(1)}$ dan kovarian $\boldsymbol{\Sigma}^{(1)}$ akan digunakan pada step-I untuk

menghasilkan satu set imputasi data lengkap yang baru dan dapat dituliskan sebagai berikut:

$$Y_{mis}^{(2)}|Y_{obs} \sim \mathbf{N_q}\left(\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)}\right)$$

Setelah nilai imputasi untuk data hilang yaitu $Y_{mis}^{(2)}$ diperoleh, nilai vektor mean dan matriks kovarian akan kembali diperbaharui. Seperti pada iterasi $t = 1$, pada step-P akan disimulasi kembali nilai posterior untuk masing-masing parameter tersebut.

Nilai posterior parameter $\boldsymbol{\Sigma}$ pada iterasi kedua diperoleh dengan cara menarik sebuah nilai posterior $\boldsymbol{\Sigma}^{(2)}$ dari distribusi bersyarat $\boldsymbol{\Sigma}^{(2)}$ yaitu jika nilai data Y_{obs} dan $Y_{mis}^{(2)}$ diketahui. Distribusi bersyaratnya adalah distribusi invers wishart dan dapat dituliskan sebagai berikut:

$$\boldsymbol{\Sigma}^{(2)}|\boldsymbol{\mu}^{(2)}, Y_{obs}, Y_{mis}^{(2)} \sim W^{-1}(n-1, (n-1)S)$$

Sama seperti pada tahap posterior di iterasi sebelumnya, simulasi monte carlo akan digunakan untuk menarik sebuah nilai matriks kovarian yang baru dari distribusi posterior untuk $\boldsymbol{\Sigma}^{(2)}$. Dan nilai untuk $(n-1)S$ yaitu matriks cssp pada setiap iterasi juga akan berubah, karena nilai mean untuk setiap iterasi akan selalu diperbaharui

Nilai posterior vektor mean, dapat disimulasi dengan cara menarik sebuah nilai posterior $\boldsymbol{\mu}^{(2)}$ dari distribusi bersyarat $\boldsymbol{\mu}^{(2)}$ jika nilai $Y_{obs}, Y_{mis}^{(2)}$ dan nilai $\boldsymbol{\Sigma}^{(2)}$ diketahui. Distribusi bersyaratnya adalah distribusi normal, dapat dituliskan sebagai berikut:

$$\boldsymbol{\mu}^{(2)}|(Y_{obs}, Y_{mis}^{(2)}, \boldsymbol{\Sigma}^{(2)}) \sim N\left(\bar{y}, \frac{1}{n}\boldsymbol{\Sigma}^{(2)}\right)$$

dari iterasi $t = 2$ akan diperoleh nilai $\left(Y_{mis}^{(2)}, \left(\boldsymbol{\Sigma}^{(2)}, \boldsymbol{\mu}^{(2)}\right)\right)$.

Step-I dan step-P ini akan terus berulang hingga $t = 1,2, \ldots$ dan akan diperoleh rangkaian:

$$\left(Y_{mis}^{(1)}, \left(\boldsymbol{\Sigma}^{(1)}, \boldsymbol{\mu}^{(1)}\right)\right), \left(Y_{mis}^{(2)}, \left(\boldsymbol{\Sigma}^{(2)}, \boldsymbol{\mu}^{(2)}\right)\right), \ldots$$

yang konvergen kepada distribusi $P\left((\boldsymbol{\mu}, \boldsymbol{\Sigma}), Y_{mis}|Y_{obs}\right)$.

Step-I dan step-P pada DA akan dilakukan secara berulang sampai diperoleh nilai-nilai data hilang yang konvergen kepada distribusi yang dituju yaitu distribusi $p(Y_{mis}, \theta|Y_{obs})$.

3. Data dan Metode

3.1. Data

Contoh penerapan data hilang yang akan diolah dengan algoritma DA adalah data kebugaran fisik dari pria yang mengikuti kursus kebugaran fisik di N.C. State University. Data ini diproleh dari menu Help & Documentation SAS 9.1[8]. Data kebugaran tersebut terdiri atas variabel *Oxygen* (tingkat asupan, ml per kilogram berat tubuh per menit), *RunTime* (waktu tempuh 1.5 mil per menit), dan *RunPulse* (jumlah denyut jantung per menit). Pada ketiga variabel data kebugaran tersebut secara acak ditemukan adanya nilai data yang hilang. Data lengkapnya dapat dilihat di Lampiran.

3.2. Metode

Untuk melakukan pendugaan nilai data hilang dengan menggunakan DA, diperlukan langkah-langkah sebagai berikut:

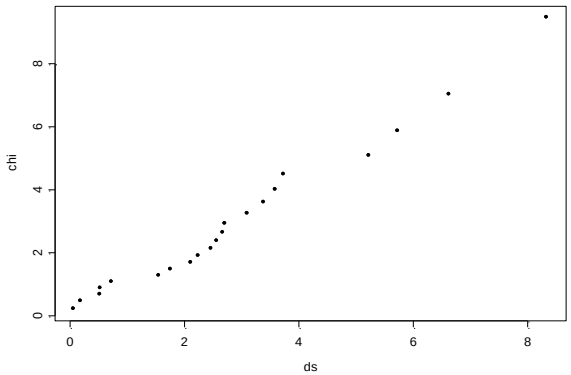
1. Menguji distribusi dari data, dalam hal ini diharapkan data berasal dari distribusi normal multivariate.
2. Memberikan nilai awal bagi parameter $\boldsymbol{\mu}$ dan $\boldsymbol{\Sigma}$.
3. Menentukan *burn in period* dengan memperhatikan nilai autokorelasi dan grafik *trace plot*.
4. Menghitung dugaan terhadap nilai data yang hilang.

Langkah 2 sampai dengan 4 dilakukan dengan memanfaatkan software SAS 9.1.

4. Hasil dan Pembahasan

Sebelum nilai data yang hilang diduga, terlebih dahulu dilakukan uji terhadap distribusi dari data dengan tidak melibatkan seluruh data yang hilang. Dalam hal ini digunakan uji Kolmogorov-Smirnov dengan taraf signifikansi 5%. Berdasarkan output *Kolmogorov-Smirnov* dapat diketahui bahwa nilai statistik ujinya sebesar 0,1192, dengan nilai p-value sebesar 0,8927. Karena nilai p-value lebih besar dari α = 5%, maka dapat disimpulkan bahwa data berasal dari berdistribusi normal multivariat. Gambar 1 dibawah ini menampilkan garfik Q-Q plot dari data. Tampak bahwa titik-titik data cenderung membentuk pola garis lurus. Hal ini mengindikasikan bahwa tidak ada alasan untuk menolak bahwa data berasal dari distribusi normal multivariate.

Untuk menduga nilai hilang pada data tersebut digunakan bantuan software SAS 9.1. Sotware ini mengggunakan metode DA untuk mengimputasi nilai-nilai hilang dari variabel yang diamati. Secara baku, prosedur pada SAS 9.1 menggunakan sebuah rantai tunggal untuk menciptakan 5 imputasi. Pada awalnya akan dilakukan iterasi sebanyak 200 iterasi sebelum menetapkan nilai untuk data hilang pada imputasi pertama. Proses iterasi awal ini merupakan proses *burn-in period*, yaitu proses untuk menghilangkan pengaruh nilai awal. Distribusi dari iterasi yang terbaru akan mendekati distribusi posterior. Setelah proses *burn-in period*, dilanjutkan dengan proses imputasi berikutnya yang terdiri atas 100 iterasi untuk setiap imputasi. Nilai yang diambil untuk nilai data hilang Y_{mis} pada tiap satu imputasi adalah nilai akhir dari iterasi pada setiap imputasi tersebut.



Gambar 1. Plot Q-Q Normal Multivariat

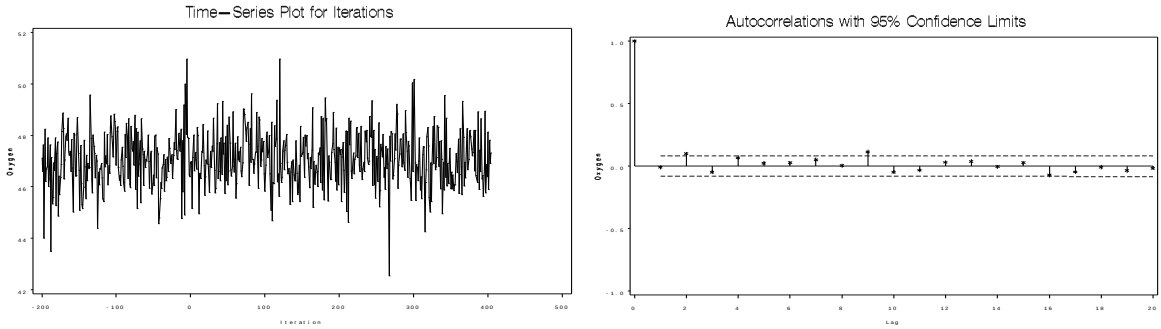
Untuk melakukan pendugaan terhadap nilai data hilang, pada iterasi pertama diberikan nilai awal untuk parameter vektor mean dan matrik kovarian yaitu $\left(\boldsymbol{\mu}^{(0)}, \boldsymbol{\Sigma}^{(0)}\right)$. Nilai awal untuk mean dan kovarian pada program SAS 9.1 diperoleh dari nilai maksimal

posterior berdasarkan algoritma *Expectation Maximization* (EM). Nilai awal tersebut adalah

$$\boldsymbol{\mu}^0 = \begin{pmatrix} 47{,}103 \\ 10{,}554 \\ 171{,}382 \end{pmatrix} \text{ dan } \boldsymbol{\Sigma}^0 = \begin{pmatrix} 24{,}549 & -5{,}726 & -15{,}926 \\ -5{,}726 & 1{,}781 & 3{,}124 \\ -15{,}926 & 3{,}124 & 83{,}164 \end{pmatrix}$$

Dari output SAS 9.1, dihasilkan plot *time-series* dan plot autokorelasi yang menampilkan rata-rata setiap variabel pengamatan pada setiap iterasi. Plot *time-series* dan plot autokorelasi berguna untuk memantau konvergensi dari proses data augmentation. Berikut ini akan ditampilkan plot *time-series* dan plot autokorelasi untuk masing-masing variabel pengamatan.

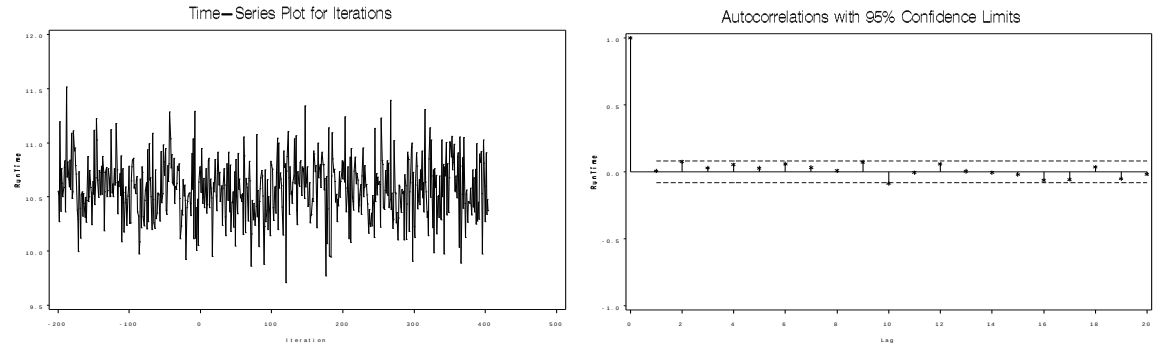
Plot *time-series* dan plot autokorelasi untuk rata-rata variabel pengamatan *Oxygen* setelah nilai data yang hilang diduga pada setiap iterasi adalah:



(a) Plot *Time Series Mean Oxygen* (b) Plot Autokorelasi *Mean Oxygen*

Gambar 2. Plot untuk Pengujian Konvergensi *Oxygen*

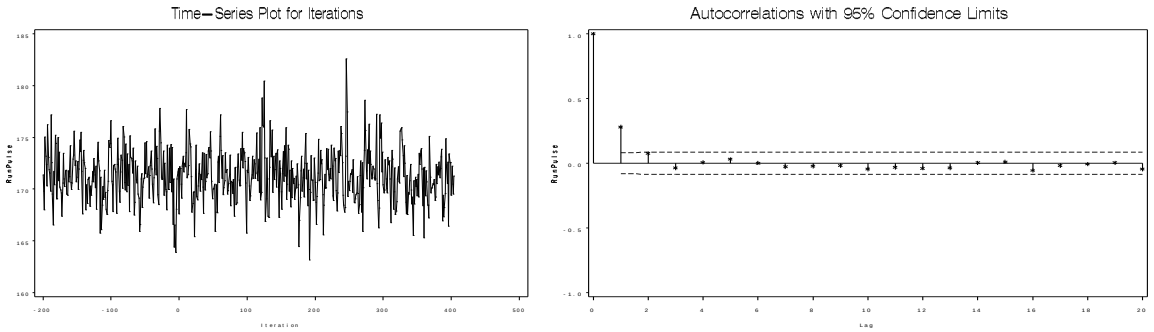
Plot *time-series* dan plot autokorelasi untuk rata-rata variabel pengamatan *RunTime* setelah nilai data yang hilang diduga pada setiap iterasi adalah:



(a) Plot *Time Series Mean RunTime* (b) Plot Autokorelasi *Mean RunTime*

Gambar 3. Plot untuk Pengujian Konvergensi *RunTime*

Plot *time-series* dan plot autokorelasi untuk rata-rata variabel pengamatan RunPulse setelah nilai data yang hilang diduga pada setiap iterasi adalah:



(a) Plot *Time Series Mean RunPulse* (b) Plot Autokorelasi *Mean RunPulse*

Gambar 4. Plot untuk Pengujian Konvergensi *RunPulse*

Secara keseluruhan plot *time-series* untuk variabel pengamatan *Oxygen*, *RunTime* dan *RunPulse* menunjukkan tidak adanya trend naik atau turun untuk masing-masing variabel pengamatan tersebut. Sehingga dapat disimpulkan bahwa rata-rata dari variabel pengamatan *Oxygen*, *RunTime*, dan *RunPulse* setelah proses imputasi adalah konvergen.

Plot autokorelasi untuk variabel pengamatan *Oxygen*, *RunTime* dan *RunPulse* merupakan plot autokorelasi yang menampilkan rata-rata setiap variabel pengamatan setelah proses pendugaan data hilang dilakukan. Plot menunjukkan tidak adanya autokorelasi positif atau negatif yang signifikan, yang berarti bahwa rata-rata hasil dugaan data hilang untuk setiap variabel pengamatan adalah tidak berautokorelasi. Hal ini menunjukkan bahwa rata-rata setiap variabel pengamatan bergerak stabil kepada dan konvergen kepada distribusi posterior yang dituju.

Jumlah imputasi yang dijalankan untuk proses pendugaan nilai data hilang pada makalah ini sebanyak m = 5 imputasi. Untuk setiap variabel pengamatan akan diambil masing-masing satu nilai $Y_{i(mis)}$yang terletak di urutan terakhir iterasi pada setiap imputasi. Sehingga akan diperoleh 5 nilai $Y_{i(mis)}$ yang kemudian akan dirata-ratakan. Rata-rata dari 5 nilai $Y_{i(mis)}$ tersebutlah yang kemudian akan menjadi nilai dugaan akhir pada setiap kasus data hilang yang terjadi di setiap variabel. Berikut ini akan diberikan masing-masing dugaan nilai data hilang yang terjadi pada variabel *Oxygen*, *RunTime*, dan *RunPulse*.

Untuk variabel Oxygen, kasus nilai data hilang terjadi pada 3 observasi, yaitu pada observasi ke 4, 20 dan 28. Tabel 1 adalah nilai dugaan data hilang hasil dari 5 imputasi untuk setiap observasi tersebut.

Tabel 1. Imputasi Nilai Data Hilang pada Variabel *Oxygen*

m	*Oxygen*		
	i = 4	i = 20	i = 28
1	45,8453	42,6682	48,9234
2	44,4772	48,7259	53,3759
3	46,0266	49,8432	52,5534
4	38,9083	44,0570	54,2722
5	41,2781	45,2122	51,4256
$\overline{Y}_{i,mis}$	43,3071	46,1013	52,1101

Nilai dugaan yang digunakan untuk mengisi nilai data hilang pada observasi ke 4, 20, dan 28 untuk variabel *Oxygen* adalah rata-rata nilai dugaan dari seluruh imputasi. Sehingga dapat diperoleh nilai dugaan untuk setiap nilai data hilang, yaitu untuk observasi ke 4 adalah 43,3071, untuk observasi ke 20 adalah 46,1013, dan untuk observasi ke 28 adalah 52,1101.

Untuk variabel *RunTime*, kasus nilai data hilang terjadi pada 3 observasi, yaitu pada observasi ke 6, 18, dan 23. Berikut ini akan ditampilkan nilai dugaan data hilang hasil dari 5 imputasi untuk setiap observasi tersebut.

Tabel 2. Imputasi Nilai Data Hilang pada Variabel *RunTime*

m	*RunTime*		
	i = 6	i = 18	i = 23
1	9,6678	6,4514	11,7947
2	10,8393	9,4047	11,8769
3	10,3526	6,3690	12,0637
4	9,9595	7,8424	10,5306
5	9,8532	7,4920	9,7734
$\overline{Y}_{i,mis}$	10,1345	7,5119	11,2079

Nilai dugaan yang digunakan untuk mengisi nilai data hilang pada observasi ke 6, 18, dan 23 untuk variabel *RunTime* adalah rata-rata nilai dugaan dari seluruh imputasi. Sehingga nilai dugaan untuk observasi ke 6 adalah 10,1345, untuk observasi ke 18 adalah 7,5119, dan untuk observasi ke 23 adalah 11,2079.

Untuk variabel pengamatan *RunPulse*, kasus data hilang terjadi pada 9 observasi, yaitu pada observasi ke 3, 6, 10, 13, 18, 20, 23, 25 dan 28. Nilai dugaan yang digunakan untuk mengisi nilai data hilang pada observasi ke 3, 6, 10, 13, 18, 20, 23, 25 dan 28 untuk variabel *RunPulse* adalah adalah rata-rata nilai dugaan dari seluruh imputasi. Sehingga nilai dugaan untuk observasi ke 3 = 172,844, observasi ke 6 = 166,919, observasi ke 10 = 168,199, observasi ke 13 = 168,132, observasi ke 18 = 151,466, observasi ke 20 = 171, 272, observasi 23 = 173, 692, observasi ke 25 = 169,456, dan untuk observasi ke 28 = 162,839. Nilai dugaan untuk sembilan observasi tersebut akan menyebabkan data variabel *RunPulse* menjadi data yang lengkap. Berikut ini akan ditampilkan nilai dugaan data hilang hasil dari 5 imputasi untuk setiap observasi tersebut.

Tabel 3. Imputasi Nilai Data Hilang pada Variabel *RunPulse*

m	*RunPulse*								
	i = 3	i = 6	i = 10	i = 13	i = 18	i = 20	i = 23	i = 25	i = 28
1	173,152	153,736	167,663	168,475	157,713	173,367	166,679	172,442	164,467
2	177,625	173,247	156,654	167,497	153,450	166,869	171,238	161,765	161,878
3	182,673	158,908	174,895	166,419	143,389	156,708	169,385	164,113	159,272
4	170,866	180,711	172,110	149,841	145,778	177,875	190,122	175,069	169,409
5	159,904	167,94	169,675	188,430	156,999	181,542	171,036	173,889	159,168
$\overline{Y}_{i,mis}$	172,844	166,919	168,199	168,132	151,466	171,272	173,692	169,456	162,839

Dengan diperolehnya nilai dugaan untuk masing-masing data hilang, maka data pengamatan kebugaran fisik dari pria yang mengikuti kursus kebugaran fisik di N.C. State University, akan menjadi data pengamatan yang lengkap.

5. Kesimpulan

Masalah nilai data hilang pada sekumpulan data pengamatan tidak harus diatasi dengan penghapusan nilai data hilang tersebut. Nilai data hilang dapat diatasi dengan melakukan pendugaan. Proses pendugaan nilai data hilang dapat dilakukan dengan menggunakan algoritma *Data Augmentation*. Algoritma *Data Augmentation* terdiri atas dua tahap, yaitu tahap Imputasi dan tahap Posterior yang akan dilakukan secara berulang sampai diperoleh nilai-nilai data hilang yang konvergen kepada distribusi yang dituju.Kekonvergenan *Data Augmentation* dapat dilihat dari tampilan grafis yang dihasilkan pada output, yaitu plot time-series dan plot autokorelasi. Penilaian kekonvergenan berguna untuk mengetahui bahwa distribusi dari setiap iterasi yang dihasilkan adalah mendekati distribusi posterior yang benar. Nilai dugaan akhir untuk setiap data hilang adalah nilai rata-rata dari nilai dugaan pada setiap imputasi. Dimana setiap imputasi terdiri dari banyak iterasi yang dijalankan, dan diambil satu nilai dugaan untuk data hilang pada iterasi terakhir di setiap imputasi.

DAFTAR PUSTAKA

1. Casella G. and Berger R.L., *Statistical Inference*, Thomson Learning, Duxbury, 2002.
2. Howell, D.C., The analysis of missing data. In Outhwaite, W. & Turner, S. *Handbook of Social Science Methodology*, Sage, London, 2008.
3. Johnson, V.E. and Albert, J.M., *Ordinal Data Modelling*, Springer-Verlag, New York, 1998.
4. Little, R.J.A. and Rubin, D.B., *Statistical Analysis with Missing Data*. New York, John Wiley & Sons, 1987.
5. Rubin, D. B., *Multiple Imputation for Nonresponse in Surveys*, New York, John Wiley & Sons, 1987.
6. Schafer, J.L., *Analysis of Incomplete Mutivariate Data*, New York, Chapman and Hall, 1997.
7. Tanner, M.A and Wong, W.H., The Calculation of Posterior Distribution by Data Augmentation. *Journal of the American Statistical Association*, 1987, Vol.82, No. 398: 528-540.
8. SAS 9.1 Help & Documentation

Lampiran

Data kebugaran fisik yang diperoleh dari menu Help & Documentation SAS 9.1.

Nomor	*Oxygen*	*RunTime*	*RunPulse*
1	44.609	11.37	178
2	54.297	8.65	156
3	49.874	9.22	-
4	-	11.95	176
5	39.442	13.08	174
6	50.541	-	-
7	44.754	11.12	176
8	51.855	10.33	166
9	40.836	10.95	168
10	46.774	10.25	-
11	39.407	12.63	174
12	45.441	9.63	164
13	45.118	11.08	-
14	45.790	10.47	186
15	48.673	9.40	186
16	47.467	10.50	170
17	45.313	10.07	185
18	59.571	-	-
19	44.811	11.63	176
20	-	10.85	-
21	60.055	8.63	170
22	37.388	14.03	186
23	47.273	-	-
24	49.156	8.95	180
25	46.672	10.00	-
26	50.388	10.08	168
27	46.080	11.17	156
28	-	8.92	-
29	39.203	12.88	168
30	50.545	9.93	148
31	47.920	11.50	170