 e-ISSN 2598-9596

Memanfaatkan R untuk Preprocessing Data yang Efisien dalam Analisis Prediktif

[*Leveraging R for Efficient Data Preprocessing in Predictive Analytics*]

I Gusti Agung Anom Yudistira[1*]
[1]Departemen Statistika, Universitas Bina Nusantara, Jl. K H. Syahdan No. 9, Kelurahan Kemanggisan, Kecamatan Palmerah, Jakarta Barat 11480
*Korespondensi penulis: i.yudistira@binus.ac.id

ABSTRACT

The digital era has triggered a data explosion that demands efficient preprocessing capabilities. The R programming language, supported by a wide array of packages, offers effective solutions for data preprocessing, particularly in handling missing values. This study aims to demonstrate the use of R to improve the quality of predictive models and provide practical guidance for academics and practitioners. A descriptive-exploratory methodology is employed through a case study involving data preprocessing in R. The workflow includes data collection, cleaning and transformation, result visualization, and step-by-step documentation as a practical guide. In this study, simulated data were constructed by taking clean big data and then artificially injecting missing values using the R package messy. The imputation process begins with analyzing variable correlations and distributions using scatter plot matrices and histograms, followed by selecting appropriate imputation methods such as linear regression, mean, or median. R facilitates this process through comprehensive functions and visualizations. Evaluation is conducted by comparing the distribution patterns of the original data and the cleaned simulated data. The results indicate that both datasets exhibit statistically similar distribution shapes, suggesting that the imputation methods preserve the original data characteristics effectively..

Keywords : *Data mining; Data preprocessing; Predictive analysis; R programming.*

ABSTRAK

Era digital menghasilkan ledakan data yang menuntut kemampuan preprocessing data yang efisien. Bahasa R, dengan berbagai paket pendukungnya, menawarkan solusi efektif untuk preprocessing, khususnya dalam penanganan missing values. Penelitian ini bertujuan mendemonstrasikan pemanfaatan R untuk meningkatkan kualitas model prediktif dan memberikan panduan praktis bagi akademisi serta praktisi. Metodologi yang digunakan dalam penelitian adalah metode deskriptif eksploratif dengan studi kasus menggunakan R untuk preprocessing data. Tahapannya meliputi pengumpulan data, pembersihan dan transformasi data, visualisasi hasil, serta dokumentasi langkah-langkah sebagai panduan praktis. Pada penelitian ini dilakukan percobaan dengan membangun data simulasi, yang dihasilkan dari data besar yang sudah bersih, kemudian dibuat dibuat menjadi data yang tidak lengkap dengan memanfaatkan paket R `messy`. Proses imputasi data dengan R dimulai dari analisis korelasi dan distribusi variabel menggunakan *scatter plot matrix* dan histogram, memilih metode imputasi yang sesuai seperti regresi linear, rata-rata, atau median. R memudahkan proses ini lewat fungsi dan visualisasi yang lengkap. Hasil evaluasi dilakukan dengan membandingkan bentuk sebaran data asli, dengan data simulasi yang telah dibersihkan. Hasil yang diberikan menunjukkan bahwa kedua data memberikan bentuk sebaran yang tidak signifikan.

Kata kunci *:* *Analisis prediktif ; Data mining; Data praprosesing.; R programming*

e-ISSN 2598-9596

PENDAHULUAN

Dalam era digital yang ditandai dengan maraknya transaksi melalui internet dan menjamurnya konten-konten sosial media, menimbulkan ledakan data (Rahmah et al. 2025). Sehingga kemampuan untuk mengolah dan meng-analisis data secara efisien menjadi semakin penting (Reyhan et al., 2024). Salah satu tahapan krusial dalam analisis data adalah prapemrosesan data, yaitu proses pembersihan, transformasi, dan penyiapan data sebelum digunakan dalam model prediktif (Daniswara & Nuryana, 2023). Tahapan ini sangat menentukan kualitas hasil analisis, terutama dalam konteks *machine learning* dan *data mining*.

Bahasa pemrograman R dikenal luas di kalangan akademisi dan praktisi data karena kemampuannya yang kuat dalam manipulasi data, visualisasi, dan analisis statistik (Pavlenko et al., 2022). Dengan dukungan berbagai paket seperti `dplyr`, `tidyr`, dan `caret`, R menyediakan pendekatan yang fleksibel dan efisien untuk melakukan prapemrosesan data. Namun, masih banyak pengguna yang belum memanfaatkan potensi R secara optimal dalam tahap ini (Hirsch, 2023). Penanganan data hilang (*missing value*), merupakan masalah penting dalam prapemrosesan data.

Masalah yang ingin diangkat dalam penelitian ini dirumuskan sebagai berikut: Bagaimana efektivitas metode imputasi berbasis R dalam mempertahankan karakteristik distribusi data setelah proses pembersihan dibandingkan dengan data asli?

Berdasarkan rumusan masalah tersebut, tujuan penelitian adalah sebagai berikut: 1) Mendokumentasikan kode R, yang dapat dimanfaatkan secara praktis dan efisien dalam prapemrosesan data khususnya untuk melakukan pembersihan data, dalam hal ini menangani *missing values*. 2) Mendemontrasikan langkah-langkah prapemrosesan data menggunakan R untuk meningkatkan akurasi model prediktif.

Penelitian ini diharapkan memberikan manfaat sebagai berikut:

a) Menyediakan panduan praktis bagi akademisi dan praktisi dalam menggunakan R untuk preprocessing data.
b) Meningkatkan efisiensi dan kualitas proses analisis prediktif melalui penerapan teknik prapemrosesan yang tepat.
c) Mendorong pemanfaatan R sebagai alat bantu utama dalam pengolahan data di berbagai bidang aplikasi, khususnya pendidikan dan riset.

e-ISSN 2598-9596

BAHAN DAN METODE

Bahan dan Alat

Penelitian ini menggunakan perangkat lunak dan data sebagai bahan utama dalam pengolahan prapemrosesan dan analisis prediktif. Adapun bahan dan alat yang digunakan adalah sebagai berikut:

Perangkat Lunak:

a) R versi: 4.5.1
b) RStudio sebagai lingkungan pengembangan
c) Paket R yang digunakan: `tidyverse` dan `messy`.

Data

a) Dataset publik yang tersedia secara bebas untuk keperluan pembelajaran dan eksperimen analisis data. Sumber data diperoleh dari sumber publik yaitu, *Kaggle*: platform berbagi dataset dan kompetisi data science. https://www.kaggle.com/
b) Dataset yang digunakan bersifat tabular, mencakup variabel numerik dan kategorikal, serta dipilih berdasarkan kesesuaian dengan tujuan analisis prediktif dan kebutuhan prapemrosesan.

Perangkat Keras:

a) Laptop / PC dengan spesifikasi minimum: prosesor Intel i5, RAM 8 GB, dan sistem operasi Windows 11.

Metode Penelitian

Penelitian ini bersifat deskriptif-eksploratif dengan pendekatan studi kasus. Tujuannya adalah untuk mengevaluasi efektivitas metode imputasi berbasis R dalam mempertahankan karakteristik distribusi data asli setelah proses pembersihan.

Eksplorasi dilakukan untuk mengukur:

a) Efisiensi: seberapa cepat dan praktis metode dapat diterapkan.
b) Akurasi: sejauh mana hasil imputasi mendekati data asli.
c) Stabilitas: konsistensi hasil terhadap variasi data yang hilang.

Langkah-langkah Penelitian:

1. Pengumpulan Data**:** Dataset publik dari Kaggle digunakan sebagai data bersih awal / asli.
2. Simulasi *Missing Values***:** Data bersih diubah menjadi tidak lengkap menggunakan paket `messy`.
3. Imputasi Data**:** Dilakukan dengan tiga pendekatan yaitu *mean*, *median*, regresi linear, sesuai karakteristik variabel.
4. Evaluasi Hasil**:** Dibandingkan distribusi dan statistik deskriptif antara data asli dan data hasil imputasi.
5. Visualisasi**:** Histogram dan scatter plot digunakan untuk menunjukkan perubahan distribusi dan hubungan antar variabel.

e-ISSN 2598-9596

Setiap langkah dicatat dan dijelaskan secara sistematis untuk membentuk panduan praktis bagi pengguna R dalam prapemrosesan data, khususnya imputasi data.

HASIL DAN PEMBAHASAN

Pendekatan R dalam prapemrosesan data dinilai lebih efisien dan fleksibel dibandingkan perangkat lunak lain seperti SPSS, Excel, dan Python, terutama untuk kebutuhan akademik dan pembelajaran. Dengan ekosistem paket terbuka seperti tidyverse, caret, dan messy, R mendukung manipulasi data, visualisasi, dan imputasi secara terintegrasi. Keunggulan sintaks statistik yang ringkas dan visualisasi berbasis ggplot2 menjadikan R pilihan ideal untuk studi eksploratif, replikatif, dan pedagogis di lingkungan pendidikan tinggi dan riset terbuka. Berikut ini adalah tahapan-tahapan imputasi data dengan menggunakan R (Hirsch, 2023).

Penyiapan Data

Dataset "Student Performance.csv" diunduh pada link **Student Performance Dataset**, yang tersedia pada laman kaggle.com yang di bagikan oleh Ghulam Muhammad Nabeel, dan diperbarui terakhir bulan Agustus 2025. Penelitian yang juga berkaitan dengan dataset "student performance" dilakukan oleh Hasan, R., et al. (2021). Dataset ini berisi 1.000.000 baris data kinerja siswa yang realistis, dirancang khusus untuk pemula dalam bidang Machine Learning untuk berlatih regresi linear, pelatihan model, dan teknik evaluasi. Variabel-variabel data tersebut adalah: 1) `student_id`; 2) `weekly_self_study_hours`; 3) `attendance_percentage`; 4) `class_participation`; 5) `total_score`; dan 6) `grade`. Data ini terstruktur dan sudah bersih, tetapi untuk tujuan penelitian, data tersebut dibuat menjadi tidak lengkap (mengandung data hilang), yaitu dengan menggunakan paket `messy` (Rennie & Davison, 2025).

Pertama-tama data asli dibaca dan dimuat ke R. Kode R untuk membaca data asli diberikan oleh Gambar 1 berikut ini:

```
> student <- read.csv("student_performance.csv")
> dim(student)
[1] 1000000       6
```

Gambar 1. Penyiapan data

Gambar 1 memperlihatkan perintah R dibedakan dengan luarannya berdasarkan warna. Warna biru adalah perintah R dan luarannya berwarna hitam. File data dengan format .csv yang diunduh dari *kaggle.com*. File ini dibaca dengan fungsi `read.csv`, kemudian disimpan pada objek dengan nama `student`. Fungsi `dim()` memberikan ukuran baris dan kolom dari dataset `student`, yaitu 1000000 x 6. Dataset `student` adalah data yang sudah bersih, akan tetapi untuk tujuan penelitian ini, data tersebut kemudian dibuat menjadi

e-ISSN 2598-9596

tidak lengkap. Data ini dinamakan dengan data simulasi. Berbasiskan data simulasi ini kemudian dilakukan berbagai percobaan pembersihan data, dengan berbagai teknik yang ada. Kemudian kode R dibuat dan didokumentasikan, yang berguna untuk panduan pada pekerjaan serupa kedepannya. Data yang telah dibersihkan ini akan dibandingkan dengan dengan data `student` yang asli. Gambar 2 berikut adalah kode R untuk proses pembangkitan data simulasi dengan menggunakan paket `messy`.

```
> library(messy)
> stud_dirt <- student |>
+   make_missing(cols= c("attendance_percentage"), messiness = 0.1) |>
+   make_missing(cols= c("class_participation"),   messiness = 0.07) |>
+   make_missing(cols= c("weekly_self_study_hours"),   messiness = 0.02)
```

Gambar 2. Penambahan nilai `NA` pada dataset `student`

Pembersihan Data – Mengatasi *Missing Value*

Gambar 2 adalah kode R untuk menghasilkan dataset baru dengan nama `stud_dirt`. Data set ini mengandung nilai `NA` pada tiga variabelnya, yaitu `weekly_self_study_hours`, nilai `NA` muncul secara acak sebanyak 0.2% dari keselurahan datanya, *`attendance_percentage`* sebanyak 10% dan *`class_participation`*, sebanyak 0.7%. Proses ini dibangkitkan dengan menggunakan fungsi `make_missing`, yang disediakan oleh paket `messy`.

```
> summary(stud_dirt)
   student_id      weekly_self_study_hours attendance_percentage
 Min.   :      1   Min.   : 0.00           Min.   : 50.00
 1st Qu.: 250001   1st Qu.:10.30           1st Qu.: 78.30
 Median : 500001   Median :15.00           Median : 85.00
 Mean   : 500001   Mean   :15.03           Mean   : 84.71
 3rd Qu.: 750000   3rd Qu.:19.70           3rd Qu.: 91.80
 Max.   :1000000   Max.   :40.00           Max.   :100.00
                   NA's   :19993 ✓         NA's   :100219 ✓
 class_participation  total_score        grade
 Min.   : 0.000       Min.   :  9.40     Length:1000000
 1st Qu.: 4.700       1st Qu.: 73.90     Class :character
 Median : 6.000       Median : 87.50     Mode  :character
 Mean   : 5.985       Mean   : 84.28
 3rd Qu.: 7.300       3rd Qu.:100.00
 Max.   :10.000       Max.   :100.00
 NA's   :69662 ✓
```

Gambar3. Ringkasan Statistik untuk dataset `stud_dirt`.

Sebelum penanganan *missing value* dilakukan, maka proses identifikasi terhadap nilai `NA` tersebut dilakukan dengan menggunakan perintah `summary()` (Hudiburgh & Garbinsky, 2020). Hasil luarannya diperlihatkan oleh Gambar 3 (tanda centang merah). Selanjutnya diperiksa hubungan korelasi diantara vaiabel-variabel pada dataset stud_dirt. Hal ini ditampilkan dengan *scatter plot matrix*, menggunakan fungsi `pairs` (Emerson et al., 2013). Oleh karena ukuran dataset yang cukup besar (1.000.000 x 6). Proses yang dilakukan fungsi `pairs` secara langsung, bisa memakan waktu yang lama. Untuk mengatasi hal itu, Sebaiknya dilakukan pengambilan sampel acak sebesar 5.000 baris saja, tanpa kehilangan informasi mengenai trend hubungan korelasional diantara variabel-variabel tersebut. Prosesnya adalah sebagai berikut,

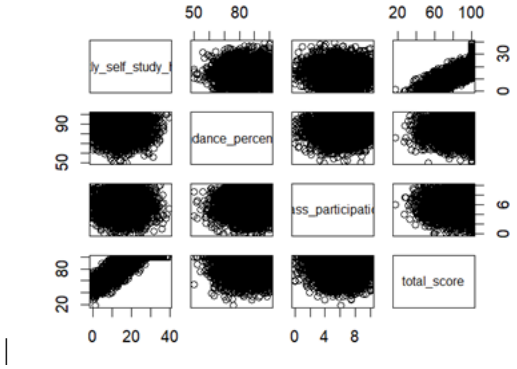
```
> set.seed(123)  # agar hasil sampling konsisten
> id <- sample(stud_dirt$student_id, size = 5000)
> student_smp <- stud_dirt[sort(id),]
> dim(student_smp)
[1] 5000    6
```

Gambar 4. Kode R pengambilan sampel berukuran 5000 baris dari dataset `stud_dirt`

e-ISSN 2598-9596

Gambar 4 menunjukkan proses pengambilan sampel acak, dengan menggunakan fungsi `sample()`. Dataset hasil pengambilan sampel ini disimpan di objek R dengan nama `student_smp`. Dimensi sampel yang diperoleh diberikan oleh fungsi `dim()`. Selanjutnya berbasis data `student_smp` ini dibangun *scatter plot matrix*. Kode R dan grafiknya diberikan oleh Gambar 5.

```
> pairs(student_smp[,-c(1,6)]
```

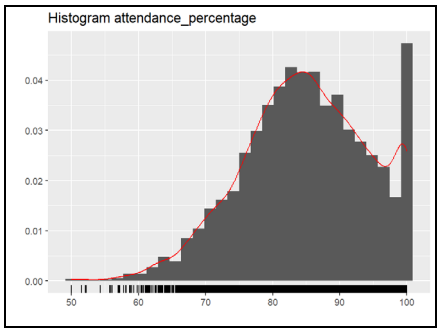


Gambar 5. *Scatter Plot Matrix*, diperoleh dengan menggunakan fungsi `pars()`

Gambar 5 mengindikasikan bahwa ketiga variabel `weekly_self_study_hours`, `attendance_percentage` dan `class_participation`, tidak saling berkorelasi. Variabel `total_score` dan `weekly_self_study_hours`, nampaknya berkorelasi cukup kuat dan positif. Jadi bisa diindikasikan bahwa *missing value* pada variabel *`weekly_self_study_hours`* dapat diduga nilainya dengan menggunakan variabel `total_score`. Variabel-variabel `attendance_percentage` dan `class_participation`, nilai *missing value* nya dapat diduga dengan nilai sentralnya (media atau rata-rata) (Lin & Tsai, 2020). Sebelum itu semua diputuskan, dilakukan pemeriksaan bentuk sebarannya terlebih dahulu, malalui histogram kedua variabel, apakah simetrik normal atau menjulur (Ochieng'Odhiambo, 2020).

```
> library(tidyverse)
> ggplot(student_smp, aes(x=attendance_percentage)) +
+   geom_histogram(aes(y=..density..))  +
+   geom_density(color="red") + geom_rug() +
+   ggtitle("Histogram attendance_percentage") +
+   xlab("") + ylab("")
`stat_bin()` using `bins = 30`. Pick better value `binwidth`.
```

Gambar 6. Kode R untuk menampilkan histogram variabel *attendance_percentage*

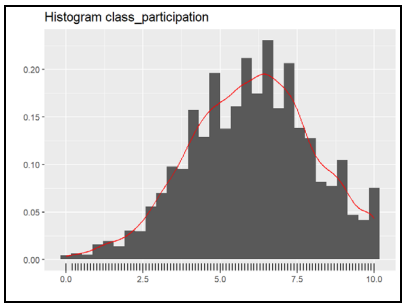


Gambar 7. Histogram variabel `attendance_percentage`

Histogram yang ditampilkan oleh Gambar 7, adalah luaran dari kode R yang ada pada Gambar 6. Histogram dari kode R Gambar 6, menggunakan paket ggplot2 yang digabungkan oleh paket `tidyverse`, dengan paket-paket lain. Untuk menampilkan histogram variabel `class_participation`, menggunakan kode R yang sama (Gambar 6), tetapi dengan mengganti variabelnya dengan

e-ISSN 2598-9596

`class_participation`. Histogramnya ditunjukkan oleh Gambar 8 berikut ini.



Gambar 8. Histogram variabel `class_participation`

Histogram untuk variabel `attendance_ percentage`, nampak menjulur ke kiri (Gambar 7). Oleh karena itu lebih tepat melakukan imputasi nilai NA pada vaiabel ini, dengan menggunakan nilai mediannya. Histogram variabel `class_participation` (Gambar 9) lebih simetrik dan berbentuk lonceng. Jadi untuk variabel `class_ participation` dapat menggunakan rata-rata (*mean*) atau median sebagai pilihan dalam imputasi *missing value*. Dataset `stud_dirt` memiliki empat variabel reguler, sehingga apabila terdapat baris dengan minimal 2 variabel dengan *missing value* (lebih dari 20%), maka sebaiknya baris tersebut dihilangkan saja. Akan tetapi harus dilakukan dengan hati-hati, dengan tetap memperhatikan berapa besar (persentase) observasi (baris) yang dihilangkan (sebaik-nya kurang dari 5%). Perhatikan kode R Gambar 9 di bawah ini, yaitu untuk melakukan tahapan-tahapan tersebut.

```
> sumNA <- apply(stud_dirt, 1, function(x) sum(is.na(x)))
> sum(sumNA >= 2)/nrow(stud_dirt) * 100
[1] 1.0125
```

Gambar 9. Kode R untuk menghitung jum-lah *missing value* setiap baris persen-tasenya

Kode R pada Gambar 9, memper-lihatkan penggunaan fungsi `apply` untuk menghitung jumlah *missing value* pada setiap baris. Ternyata jumlah baris dengan dua atau lebih NA adalah 1,0125%. Oleh karena persentase baris dengan nilai NA kurang dari 20%, maka bisa dihilangkan dari dataset, tanpa menggangu sebaran data secara keseluruhan. Proses selanjutnya adalah memangkas dataset `stud_dirt` tersebut, sehingga yang tinggal adalah baris-baris dengan maksimal satu nilai NA. Gambar 10 berikut ini memberikan kode R untuk melakukan hal tersebut.

```
> nax <- which(sumNA >= 2)
> stud_dirt1 <- stud_dirt[-nax,]; dim(stud_dirt1)
[1] 989875      6
> summary(stud_dirt1)
   student_id      weekly_self_study_hours attendance_percentage
 Min.   :      1   Min.   : 0.00           Min.   : 50.00
 1st Qu.: 250006   1st Qu.:10.30           1st Qu.: 78.30
 Median : 499993   Median :15.00           Median : 85.00
 Mean   : 499994   Mean   :15.03           Mean   : 84.71
 3rd Qu.: 750023   3rd Qu.:19.70           3rd Qu.: 91.70
 Max.   :1000000   Max.   :40.00           Max.   :100.00
                   NA's   :16786           NA's   :90961
 class_participation  total_score       grade
 Min.   : 0.000       Min.   :  9.40   Length:989875
 1st Qu.: 4.700       1st Qu.: 73.90   Class :character
 Median : 6.000       Median : 87.50   Mode  :character
 Mean   : 5.985       Mean   : 84.28
 3rd Qu.: 7.300       3rd Qu.:100.00
 Max.   :10.000       Max.   :100.00
 NA's   :61936
```

Gambar 10. Kode R untuk memangkas baris-baris dengan jumlah nilai NA maksimal satu per baris.

Kode R pada Gambar 10 ini merupakan kelanjutan dari kode R Gambar 9. Fungsi `which(sumNA >= 2)`, akan memberikan nomor-nomor baris, dengan jumlah NA lebih atau sama dengan 2, dan disimpan pada objek `nax`. Perintah

e-ISSN 2598-9596

`stud_dirt[-nax,]` akan menghilangkan baris-baris dengan nilai NA lebih atau sama dengan 2. Dataset yang baru ini, disimpan pada objek `stud_dirt1`. Terlihat juga pada Gambar 10 bahwa dimensi dari dataset `stud_dirt1` adalah 989.875 x 6. Jadi ukuran baris berkurang sebesar 1.000.000 – 989.875 = 10125. Langkah selanjutnya adalah mengisi nilai-nilai NA untuk variabel-variabel `attendance_percentage` dan `class_participation`. Langkah-langkah Kode R ditunjukkan oleh Gambar 11 berikut ini:

```
> stud_dirt1[is.na(stud_dirt1$attendance_percentage),
+  "attendance_percentage"] <- median(stud_dirt1$attendance_percentage,
+                                    na.rm=T)
> stud_dirt1[is.na(stud_dirt1$class_participation),
+            "class_participation"] <- mean(stud_dirt1$class_participation,
+                                           na.rm=T)
> summary(stud_dirt1)
   student_id      weekly_self_study_hours attendance_percentage
 Min.   :      1   Min.   : 0.00           Min.   : 50.00
 1st Qu.: 250006   1st Qu.:10.30           1st Qu.: 79.00
 Median : 499993   Median :15.00           Median : 85.00
 Mean   : 499994   Mean   :15.03           Mean   : 84.74
 3rd Qu.: 750023   3rd Qu.:19.70           3rd Qu.: 91.00
 Max.   :1000000   Max.   :40.00           Max.   :100.00
                   NA's   :16786
 class_participation  total_score       grade
 Min.   : 0.000       Min.   :  9.40   Length:989875
 1st Qu.: 4.800       1st Qu.: 73.90   Class :character
 Median : 5.985       Median : 87.50   Mode  :character
 Mean   : 5.985       Mean   : 84.28
 3rd Qu.: 7.200       3rd Qu.:100.00
 Max.   :10.000       Max.   :100.00
```

Gambar 11. Pendugaan nilai NA pada variabel `attendance_percentage` dan `class_ participation`.

Gambar 11 menunjukkan langkah-langkah kode R untuk mengisi nilai NA, dengan menggunakan nilai tengahnya yaitu median untuk variabel `attendance_percentage` dan *mean* untuk variabel `class_participation`. Sehingga variabel yang belum terisi nilai NAnya adalah `weekly_self_study_hours`. Variabel terakhir ini, sebagaimana ditunjukkan oleh grafik *scatter plot matrix* (Gambar 5), memiliki hubungan korelasional yang kuat dengan variabel `total_score`. Nilai korelasi kedua variabel ini diperoleh melalui kode R berikut:

```
> ind <- which(!is.na(stud_dirt1$weekly_self_study_hours))
> cor(stud_dirt1$weekly_self_study_hours[ind],
+     stud_dirt1$total_score[ind])
[1] 0.8121923
```

Gambar 12. Koefisien korelasi antara variabel `weekly_self_study_hours` dan `total_score`.

Gambar 12 memperlhatkan bahwa nilai korelasi antara variabel `weekly_self_study_hours` dan `total_score`, adalah 0.8121923. Nilai korelasi yang sangat kuat ini, mengindikasikan bahwa, nilai-nilai NA pada variabel `weekly_self_study_hours` dapat diduga dari variabel `total_score`, dengan menggunakan model regressi linear (Ochieng'Odhiambo, 2020). Kode R untuk mendapatkan model regressi linear dugaan, diberikan oleh Gambar 13 di bawah ini:

```
> model <- lm(weekly_self_study_hours[ind] ~ total_score[ind],
+             data = stud_dirt1)
> summary(model)

Call:
lm(formula = weekly_self_study_hours[ind] ~ total_score[ind],
    data = stud_dirt1)

Residuals:
     Min       1Q   Median       3Q      Max
-16.9739  -2.7186  -0.1496   2.4903  19.2642

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)      -1.557e+01  2.265e-02  -687.4   <2e-16 ***
total_score[ind]  3.631e-01  2.644e-04  1373.3   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.025 on 973087 degrees of freedom
Multiple R-squared:  0.6597,    Adjusted R-squared:  0.6597
F-statistic: 1.886e+06 on 1 and 973087 DF,  p-value: < 2.2e-16
```

Gambar 13. Model regressi linear dugaan antara variabel

e-ISSN 2598-9596

`weekly_self_study_hours` dan `total_score`.

Gambar 13 memberikan hubungan regressi antara `weekly_self_study_hours`, yang bertindak sebagai variabel tidak bebas dan `total_score` yang bertindak sebagai variabel bebas (Cook & Weisberg, 2009). Variabel `weekly_self_study_hours`, bertindak sebagai variabel tidak bebas, karena merupakan variabel yang akan diduga nilainya. Objek `ind` merujuk pada nomor-nomor baris, dimana nilai-nilai variabel `weekly_self_study_hours` tidak mengandung NA. Selanjutnya Gambar 14 berikut, adalah proses pendugaan nilai NA untuk variabel `weekly_self_study_hours`. Pertama-tama dicari nilai dugaan untuk *intercept* dan *slope* model regressi tersebut.

```
> intercept <- coef(model)[1]; slope <- coef(model)[2]
> c(intercept, slope)
    (Intercept) total_score[ind]
    -15.5734113        0.3630923
```

Gambar 14. *Intercept* dan *Slope* untuk garis regressi dugaan.

Pada Gambar 14 diperoleh *intercept*=-15.5734113 dan *slope*= 0.3630923. Sehingga persamaan garis lurus yang dipakai untuk menduga variabel y = `weekly_self_study_hours`, adalah sebagai berikut.

$$\hat{y} = -15.5734 + 0.3631\ total\_score \ldots (1)$$

Persamaan (1) merupakan persamaan garis lurus yang menghubungkan antara `weekly_self_study_ hours` dan `total score`.

Langkah selanjutnya adalah menduga nilia-nilai NA pada variabel `weekly_self_study_hours`, dengan menggunakan garis lurus persamaan (1).

```
> ind2 <- which(is.na(stud_dirt1$weekly_self_study_hours))
> stud_dirt1$weekly_self_study_hours[ind2] <-
+   intercept + slope * stud_dirt1$total_score[ind2]
> stud_dirt1$weekly_self_study_hours <- ifelse(stud_dirt1$weekly_self_study_
hours < 0, 0,
+        stud_dirt1$weekly_self_study_hours)
> summary(stud_dirt1)
   student_id      weekly_self_study_hours attendance_percentage
 Min.   :      1   Min.   : 0.00           Min.   : 50.00
 1st Qu.: 250006   1st Qu.:10.30           1st Qu.: 79.00
 Median : 499993   Median :15.00           Median : 85.00
 Mean   : 499994   Mean   :15.03           Mean   : 84.74
 3rd Qu.: 750023   3rd Qu.:19.70           3rd Qu.: 91.00
 Max.   :1000000   Max.   :40.00           Max.   :100.00
 class_participation  total_score       grade
 Min.   : 0.000       Min.   :  9.40   Length:989875
 1st Qu.: 4.800       1st Qu.: 73.90   Class :character
 Median : 5.985       Median : 87.50   Mode  :character
 Mean   : 5.985       Mean   : 84.28
 3rd Qu.: 7.200       3rd Qu.:100.00
 Max.   :10.000       Max.   :100.00
```

Gambar 15. Proses pengisian nilai NA variabel `weekly_self_study_hours`, dengan menggunakan persamaan garis lurus

Gambar 15 menunjukkan proses pengisian nilai NA untuk variabel `weekly_self_study_hours`, dengan pertama-tama mengidentifikasi nomor-nomor baris yang mengandung NA. Nomor-nomor baris tersebut disimpan pada objek `ind2`. Kemudian proses pengisian tersebut mengikuti persamaan (1). Oleh karena *intercept* persamaan (1) adalah negatif, maka ada kemungkinan nilai variabel `weekly_self_study_hours` adalah negatif, sedangkan nilai negatif tidak mungkin terjadi untuk variabel ini. Jadi untuk menccgah hal ini terjadi, digunakan fungsi `ifelse`. Jika nilai dugaannya negatif, maka nilainya sama dengan nol, sedangkan jika tidak negatif, maka nilai

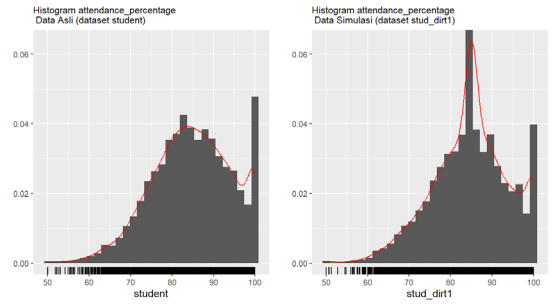
e-ISSN 2598-9596

yang ada digunakan. Perintah `summary`, memberikan luaran berupa ringkasan statistik untuk semua variabel pada dataset `stud_dirt1`. Berdasarkan luaran tersebut, nampak sudah tidak dijumpai lagi nilai NA pada semua variabel (Haliduola et al., 2022).

Pembersihan Data – Verifikasi dan Evaluasi

Dataset asli yang digunakan adalah dataset `student`. Dataset ini adalah data yang sudah bersih, tetapi kemudian dibuat data tersimulasi, dengan membuat beberapa baris pada variabel-variabel `weekly_self_study_hours`, *`attendance_percentage`*, dan `class_participation` menjadi NA (missing value). Dataset ini dibersihkan kembali, dan diberi nama dataset `stud_dirt1`.

Variabel `attendance_percentage` diimputasi dengan mediannya. Perbandingan bentuk sebaran data asli (`student`) dengan dataset tersimulasinya (`stud_dirt1`), diperlihatkan oleh histogram berikut.



Gambar 16. Perbandingan histogram variabel `attendance_percentage` dataset `student` vs `stud_dirt1`

Pada Gambar 16, terlihat bahwa histogram untuk variabel `attendance_percentage`, untuk data tersimulasi ada lonjakan pada nilai mediannya. Hal ini terjadi karena imputasi pada variabel ini menggunakan median. Akan tetapi secara umum bentuk sebaran variabel tersebut pada kedua dataset, tidak terlalu berubah secara signifikan, kecuali lonjakan pada kelas yang mengandung median. Perbandingan ringkasan statitik variabel `attendance_percentage`, pada kedua data set adalah sebagai berikut:

```
> summary(student$attendance_percentage)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  50.00   78.30   85.00   84.71   91.80  100.00
> summary(stud_dirt1$attendance_percentage)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  50.00   79.00   85.00   84.74   91.00  100.00
```

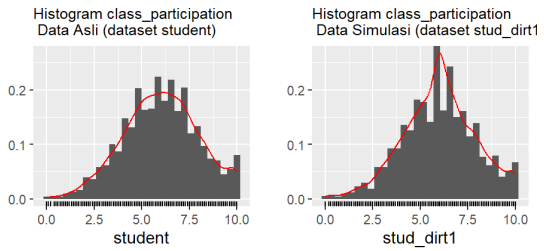
Gambar 17. Perbandingan ringkasan statistik variabel `attendance_percentage` pada dataset `student` dan `stud_dirt1`.

Gambar 17 memberikan hasil perbandingan ringkasan statistik variabel `attendance_percentage`, pada dataset `student` vs. `stud_dirt1`. Kedua

e-ISSN 2598-9596

ringkasan secara umum tidak berbeda signifikan.

Gambar 18 berikut ini memberikan perbandingan histogram variabel `class_participation` untuk dataset `student` dan `stud_dirt1`.



Gambar 18. Perbandingan histogram varia-bel `class_participation` dataset student vs stud_dirt1.

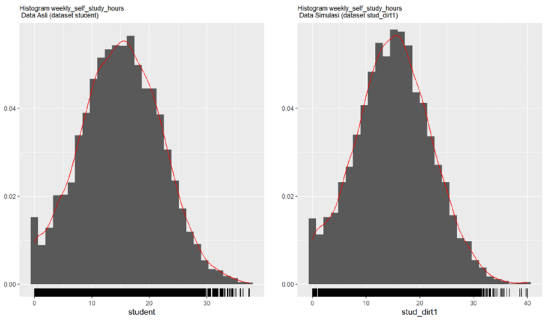
Gambar 18 memperlihatkan bentuk sebaran kedua dataset tidak berubah secara signifikan. Hal ini diperkuat dengan perbandingan ringkasan statistik variabel `class_participation` pada kedua dataset. Sebagaimana ditunjukkan oleh Gambar 19 di bawah ini:

```
> summary(student$class_participation)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   4.700   6.000   5.985   7.300  10.000
> summary(stud_dirt1$class_participation)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   4.800   5.985   5.985   7.200  10.000
```

Gambar 19. Perbandingan ringkasan statistik variabel `class_participation` pada dataset student dan stud_dirt1.

Imputasi untuk variabel `class_participation`, mengunakan rata-rata (*mean*).

Variabel `weekly_self_study_hours` diimputasi menggunakan model regressi linear. Gambar 20 memberikan perbandingan histogramnya, sedangkan Gambar 21 memberikan perbandingan ringkasan statsitiknya.



Gambar 20. Perbandingan histogram varia-bel `weekly_self_study_hours` pada dataset `student` vs `stud_dirt1`.

Histogram variabel `weekly_self_study_hours`, memiliki bentuk yang sangat serupa, yaitu simetrik dan berbentuk lonceng, dengan nilai-nilai kecil terpotong. Karena variabel ini tidak bisa bernilai negatif. Hal ini juga diperkuat oleh perbandingan ringkasan statistik keduanya, sebagaimana ditunjukkan oleh Gambar 21.

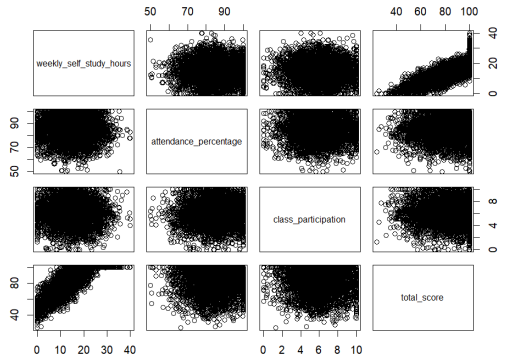
```
> summary(student$weekly_self_study_hours)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00   10.30   15.00   15.03   19.70   40.00
> summary(stud_dirt1$weekly_self_study_hours)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00   10.30   15.00   15.03   19.70   40.00
```

Gambar 21. Perbandingan ringkasan statistik variabel `weekly_self_study_hours` pada dataset student dan `stud_dirt1`.

Ringkasan statistik untuk kedua dataset adalah sama persis. Hal ini menunjukkan keunggulan dari metode regressi linear dalam mengimputasi *missing value*.

Grafik *scatter plot matrix* pada data `stud_dirt1`, diperagakan oleh Gambar 22 berikut ini:

e-ISSN 2598-9596



Gambar 22. *Scatter Plot Matrix* dataset `stud_dirt1`.

Gambar 22 jika dibandingkan dengan Gambar 5 yatu *scatter plot matrix* untuk dataset (asli) `student`, nampaknya sangat mirip.

Evaluasi Statistik Formal terhadap Hasil Imputasi

Untuk memperkuat klaim bahwa distribusi data tidak berubah secara signifikan setelah proses imputasi, dilakukan pengujian statistik formal pada tiga variabel: `attendance_percentage`, `class_participation`, dan `weekly_self_study_hours`. Pengujian meliputi uji beda rata-rata (uji-t), uji kesetaraan distribusi (Kolmogorov–Smirnov), serta analisis bentuk distribusi melalui nilai skewness dan kurtosis.

a) Variabel `attendance_percentage`: Uji-t menunjukkan perbedaan rata-rata yang signifikan secara statistik ($p = 0.041$), namun selisihnya sangat kecil (±0.03 poin), sehingga tidak bermakna secara praktis. Uji KS menunjukkan perbedaan distribusi yang signifikan ($p < 0.001$), yang terutama disebabkan oleh lonjakan nilai pada kelas median akibat metode imputasi median. Skewness dan kurtosis dari kedua dataset menunjukkan distribusi yang serupa dan mendekati normal.

b) Variabel `class_participation`: Uji-t menghasilkan p-value sebesar 0.824, menunjukkan tidak ada perbedaan signifikan antara rata-rata data asli dan hasil imputasi. Uji KS menunjukkan perbedaan distribusi yang signifikan secara statistik ($p < 0.001$), namun nilai D yang kecil dan bentuk distribusi yang hampir identik (skewness ≈ -0.11, kurtosis ≈ 2.8) menunjukkan bahwa metode imputasi *mean* berhasil mempertahankan karakteristik distribusi secara praktis.

c) Variabel `weekly_self_study_hours`: Uji-t menunjukkan tidak ada perbedaan signifikan antara rata-rata kedua dataset ($p = 0.992$), dengan selisih hanya ±0.0001. Uji KS menunjukkan perbedaan distribusi yang signifikan secara statistik ($p < 0.001$), namun nilai D yang sangat kecil (0.0035) dan kesamaan bentuk distribusi (skewness ≈ 0.09, kurtosis ≈ 2.78) menunjukkan bahwa metode imputasi regresi linear

e-ISSN 2598-9596

sangat efektif dalam mempertahankan distribusi data asli.

Secara keseluruhan, meskipun uji statistik formal menunjukkan beberapa perbedaan yang signifikan secara statistik, perbedaan tersebut sangat kecil dan tidak berdampak signifikan secara praktis. Visualisasi histogram, scatter plot matrix, dan ringkasan statistik mendukung kesimpulan bahwa metode imputasi yang digunakan berhasil mempertahankan karakteristik distribusi data secara umum.

Berdasarkan hasil semua ini nampaknya proses pembersihan data, terutama penanganan data NA, cukup berhasil, karena secara umum tidak menyimpang dengan dataset awal. Hasil simulasi ini akan menambah keyakinan dalam menggunakan metode ini untuk melakukan imputasi, seperti tahapan-tahapan yang telah dilakukkan.

Penelitian ini memiliki keterbatasan, karena bersifat eksploratif dan berfokus pada dokumentasi teknis serta demonstrasi metode imputasi menggunakan R. Evaluasi dilakukan berdasarkan pendekatan deskriptif dan visual, dengan tambahan uji statistik sederhana untuk mendukung interpretasi. Namun, penelitian ini belum mencakup pengujian inferensial yang lebih kompleks, serta belum membandingkan metode imputasi yang lebih canggih seperti kNN atau multiple imputation. Dataset yang digunakan berasal dari satu sumber (Kaggle – Student Performance), sehingga generalisasi hasil ke konteks lain masih terbatas. Selain itu, aspek efisiensi komputasi seperti waktu proses dan penggunaan memori belum dianalisis secara sistematis. Keterbatasan ini membuka peluang untuk penelitian lanjutan yang lebih mendalam dan komprehensif.

Semua kode R dalam pembahasan ini disediakan pada link di lampiran: Artikel.R (.R), Artikel.txt (.txt)

KESIMPULAN

Hasil penelitian menunjukkan bahwa metode imputasi menggunakan regresi linear dalam R memberikan hasil yang paling konsisten terhadap distribusi data asli, dibandingkan dengan metode imputasi berbasis nilai tengah seperti *mean* dan median. Pemilihan metode imputasi yang tepat dapat didasarkan pada karakteristik distribusi variabel: regresi linear untuk variabel yang berkorelasi kuat, median untuk distribusi yang menjulur, dan mean untuk distribusi simetris. Proses verifikasi melalui histogram, *scatter plot matrix*, dan ringkasan statistik mendukung efektivitas pendekatan ini. Dengan dukungan fungsi-fungsi dan visualisasi yang lengkap, R terbukti mempermudah proses prapemrosesan data secara efisien

dan terstruktur. Untuk pengembangan selanjutnya, teknik imputasi lain seperti *k-Nearest Neighbors* (kNN), serta tahapan prapemrosesan lainnya seperti integrasi, transformasi, dan reduksi data, sangat disarankan untuk diteliti lebih lanjut guna memperluas cakupan dan penerapan metode ini.

UCAPAN TERIMA KASIH

Pertama-tama penulis ucapkan terima kasih kepada *The R Core Team* (https://www.r-project.org) dan para kontri-butor yang telah memelihara dan mengem-bangkan sistem R, sehingga menjadi maju dan semakin baik kinerjanya dari setiap versi ke versi berikutnya, serta yang terpen-ting R tetap non komersial. Terima kasih juga kepada teman-teman Binus dan UPH, yang mana dalam interaksi sehari-hari, dan dalam beberapa kesempatan telah memberi-kan pelatihan R, yang membuat penulis menjadi lebih terasah dan terampil dalam menguasai sistem R.

DAFTAR PUSTAKA

Cook, R. D., & Weisberg, S. (2009). *An introduction to regression graphics*. Vol. 405. John Wiley & Sons.

Daniswara, A. A. A., & Nuryana, I. K. D. (2023). Data preprocessing pola pada penilaian mahasiswa program profesi guru. *Journal of Informatics and Computer Science (JINACS)*, *5*(1), 97–100.

Emerson, J. W., Green, W. A., Schloerke, B., Crowley, J., Cook, D., Hofmann, H., & Wickham, H. (2013). The generalized pairs plot. *Journal of Computational and Graphical Statistics*, *22*(1), 79–91. https://doi.org/10.1080/10618600.2012.694762

Haliduola, H. N., Bretz, F., & Mansmann, U. (2022). Missing data imputation using utility-based regression and sampling approaches. *Computer Methods and Programs in Biomedicine*, *226*, 107172. https://doi.org/10.1016/j.cmpb.2022.107172

Hamdani, I. M., Nurhidayat, N., Karman, A., & Julyaningsih, A. H. (2024). Edukasi dan pelatihan data science dan data preprocessing. *Intisari: Jurnal Inovasi Pengabdian Masyarakat*, *2*(1), 19–26. https://doi.org/10.58227/intisari.v2i1.125

Hasan, R., Palaniappan, S., Mahmood, S., Abbas, A., & Sarker, K. U. (2021). Dataset of students' performance using student information system, Moodle and the mobile application "eDify". *Data*, *6*(11), 110. https://doi.org/10.3390/data6110110

Hirsch, R. (2023). Introduction to R. In *Analysis of epidemiologic data using R* (pp. 1–12). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-41914-0_1

Hsu, J. L., Jones, A., Lin, J.-H., & Chen, Y.-R. (2022). Data visualization in introductory business statistics to strengthen students' practical skills. *Teaching Statistics*, *44*, 21–28. https://doi.org/10.1111/test.12291

Hudiburgh, L. M., & Garbinsky, D. (2020). Data visualization: Bringing data to life in an introductory statistics course. *Journal of Statistics Education*, *28*, 262–279. https://doi.org/10.1080/10691898.2020.1796399

Lin, W. C., & Tsai, C. F. (2020). Missing value imputation: A review and analysis of the literature (2006–2017). *Artificial Intelligence Review*, *53*(2), 1487–1509.

Ochieng'Odhiambo, F. (2020). Comparative study of various methods of handling missing data. *Mathematical Modelling and Applications*, *5*(2), 87. https://doi.org/10.11648/j.mma.20200502.14

Pavlenko, L. V., Pavlenko, M. P., Khomenko, V. H., & Mezhuyev, V. I. (2022). Application of R programming language in learning statistics. In *Proceedings of the 1st Symposium on Advances in Educational Technology* (Vol. 2, pp. 62–72). https://doi.org/10.5220/0010928500003364

Rahmah, F. R., Sutami, N. A. Z. S., Amanda, M. D. A., & Asbari, M. A. (2025). Ledakan informasi dan kesehatan mental: Peran kecerdasan emosional di era digital. *Journal of Information Systems and Management (JISMA)*, *4*(2), 19–28. https://jisma.org/index.php/jisma/article/view/1170/234

Rennie, N., & Davison, J. (2025). Making 'messy' data: An R package for teaching data wrangling with realistic data. *Teaching Statistics*. https://nrennie.rbind.io/making-messy-data/

Reyhan, M., Ahmad, D. R., Ramadhan, N. A., & Kusumasari, I. R. (2024). Penggunaan data analisis dan big data dalam strategi pengambilan keputusan keuangan. *Jurnal Akuntansi, Manajemen, dan Perencanaan Kebijakan*, *2*(2), 9. https://doi.org/10.47134/jampk.v2i2.540