1.6. Pemeriksaan dan Pengujian Bentuk Sebaran Hipotetik

Plot Kuantil-Kuantil

Sebelum mendiskusikan berbagai teknik pemeriksaan dan pengujian bentuk sebaran data dibandingkan dengan sebaran hipotetik, kita akan memulai subbab ini dengan membahas istilah persentil (*percentile*) dan kuantil (*quantile*).

Persentil ke-k dengan $0 \le k \le 100$ merupakan data yang berada pada posisi atau urutan ke $k\% \times n$, dimana n adalah ukuran contoh dan amatan diurutkan dari yang terkecil hingga yang terbesar. Misalnya saja nilai persentil ke-25 dinotasikan P25 dari sebuah gugus data berisi 400 amatan adalah data yang berada pada urutan ke-100 (diperoleh dari $25\% \times 400$). Jika seandainya diperoleh bahwa P25 bernilai 48 maka itu berarti ada 25% amatan yang nilainya kurang dari 48 dan sisanya sebanyak 75% bernilai lebih besar daripada 48.

Istilah lain yang serupa adalah nilai kuantil. Nilai kuantil biasanya diikuti dengan fraksi antara 0 dan 1, misalnya kuantil 0.25 atau dinotasikan $Q(0.25)$. Nilai $Q(t)$ dengan $0 \le t \le 1$ pada prinsipnya diperoleh sebagai data pada urutan ke $t \times n$ setelah datanya terlebih dahulu diurutkan dari yang terkecil ke yang terbesar.

Namun demikian dalam banyak algoritma komputasinya nilai kuantil diperoleh dengan proses berikut. Andaikan terdapat suatu gugus data $x_1, x_2, \ldots, x_n$. Kuantil dengan fraksi tertentu diperoleh dengan cara sebagai berikut:

- Urutkan datanya sehingga diperoleh $x_{(1)} \le x_{(2)} \le \cdots \le x_{(n)}$.
- Setiap data yang terurut merupakan kuantil yang bersesuaian dengan fraksi

  $p_i = \dfrac{i-1}{n-1}$ untuk $i = 1, \ldots, n$

- Kuantil untuk fraksi lain diperoleh dengan melakukan interpolasi linear

Sebagai ilustrasi, andaikan kita memiliki data contoh dengan 6 buah amatan sebagai berikut: {3.7, 2.7, 3.3, 1.3, 2.2, 3.1}. Setelah diurutkan datanya kita dapatkan

1.3 2.2 2.7 3.1 3.3 3.7

Selanjutnya padankan setiap nilai yang terurut dengan bilangan fraksi antara 0 dan 1 dengan jarak yang sama dan diperoleh

fraksi	0.0	0.2	0.4	0.6	0.8	1.0
nilai kuantil	1.3	2.2	2.7	3.1	3.3	3.7

Selanjutnya nilai-nilai kuantil untuk fraksi yang lain diperoleh secara interpolasi linear dari dua buah nilai kuantil yang berdekatan fraksinya. Sebagai contoh, untuk menghitung $Q(0.25)$ maka posisinya haruslah berada di antara $Q(0.2) = 2.2$ dan $Q(0.4) = 2.7$. Menggunakan interpolasi linear kita dapatkan bahwa

$$\begin{aligned} Q(0.25) &= (0.05 * Q(0.4) + 0.15 * Q(0.2)) / 0.20 \\ &= (0.05 * 2.7 + 0.15 * 2.2) / 0.20 \\ &= 2.325 \end{aligned}$$

Informasi mengenai nilai kuantil ini selanjutnya dapat digunakan untuk mengidentifikasi apakah sebaran dari data contoh yang kita miliki serupa atau mengikuti bentuk sebaran hipotetik tertentu, misalnya Normal, Gamma, atau yang lainnya. Plot Kuantil-Kuantil atau QQplot merupakan alat grafis yang dapat digunakan untuk tujuan tersebut.

Pemeriksaan ini penting misalnya ketika kita terlibat dalam analisis statistika tertentu dan mengasumsikan sebarannya normal. QQplot dapat digunakan untuk memeriksa apakah asumsi tersebut terpenuhi. Tentu saja ini hanyalah pemeriksaan secara visual dan bukanlah pembuktian analitik sehingga akan ada subjektivitas pada saat menyimpulkan. Namun demikian QQplot memungkinkan kita secara cepat menilai apakah asumsi sebaran tersebut dapat dipenuhi dan jika tidak kita dapat memperoleh gambaran titik mana yang peranannya besar dalam pelanggaran asumsi tersebut.

Pada dasarnya, QQplot merupakan plot tebaran (*scatter plot*) yang menggmbarkan dua gugus kuantil dengan kuantil yang lain. Jika kedua nilai kuantil berasal dari sebaran yang sama maka kita akan memperoleh plot tebaran yang membentuk garis lurus. Kedua kuantil yang dimaksud akan digambarkan adalah: (1) kuantil dari data contoh, dan (2) kuantil dari sebaran hipotetik yang dipadankan.

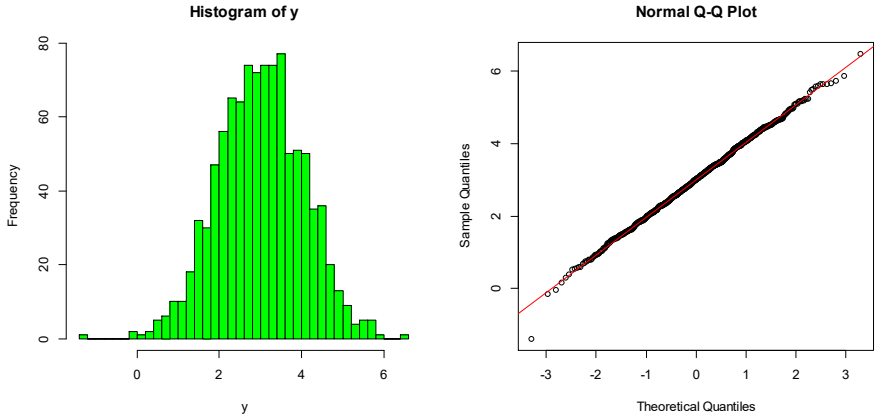
Tahapan pembuatan sebuah QQplot dari contoh berukuran n, $\{x_1, x_2, \ldots, x_n\}$, adalah sebagai berikut

- Urutkan data sehingga diperoleh susunan $x_{(1)} \leq x_{(2)} \leq \cdots \leq x_{(n)}$.
- Hitung $p_i = (i - 0.5)/n$
- Untuk sebaran hipotetik tertentu, hitung $Q_i = F^{-1}(p_i)$ dengan F adalah fungsi sebaran kumulatif. Dengan kata lain Qi adalah sebuah nilai sehingga $P(Y \leq Q_i) = p_i$
- Plot $x_{(i)}$ vs Q_i

Berdasarkan prosedur umum yang disebutkan di atas, kita dapat membuat plot kuantil-kuantil normal (*normal QQplot*) dengan tahapan sebagai berikut:

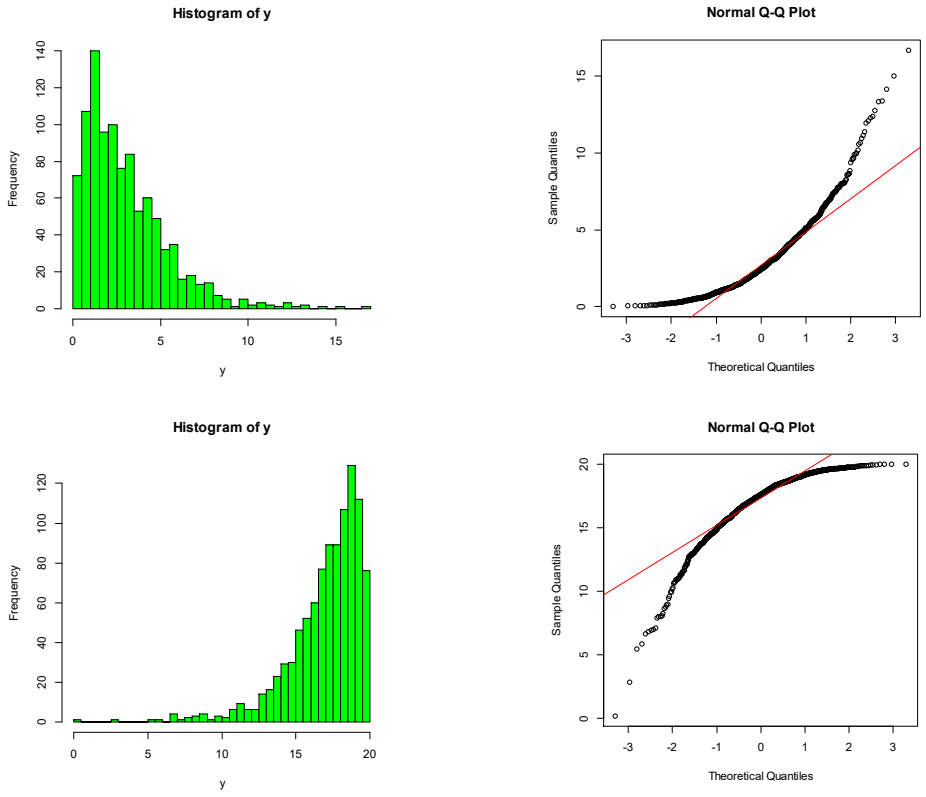
- Urutkan data $x_{(1)} \leq x_{(2)} \leq \cdots \leq x_{(n)}$
- Hitung $p_i = (i - 0.5)/n$
- Tentukan skor normal Z, untuk setiap p_i
- Plot $x_{(i)}$ vs Z_i

Jika plot kuantil-kuantil normal dari suatu data membentuk garis lurus, maka kita katakan data contoh mengikuti sebaran normal. Gambar 1.20 memberikan ilustrasi dari bentuk plot kuantil-kuantil normal dari data yang ditarik dari populasi yang menyebar normal. Terlihat bahwa plot kuantil-kuantil normalnya membentuk garis lurus yang menandakan bahwa data memang mengikuti sebaran normal.



Gambar 1.20 Histogram dan *normal Q-Q plot* untuk data contoh yang menyebar normal

Jika kita memiliki data yang tidak berasal atau tidak mengikuti sebaran normal (misalnya menjulur ke kiri atau ke kanan) maka plot kuantil-kuantil normalnya tidak akan mengikuti garis lurus. Gambar 1.21 memberikan ilustrasi bentuk plot kuantil-kuantil normal dari data menjulur ke kanan dan ke kiri. Terlihat bahwa untuk sebaran yang menjulur ke kanan bentuk plot akan menyerupai parabola yang menghadap ke atas, dan sebaliknya pada data yang menjulur ke kiri plot akan membentuk parabola menghadap ke bawah.

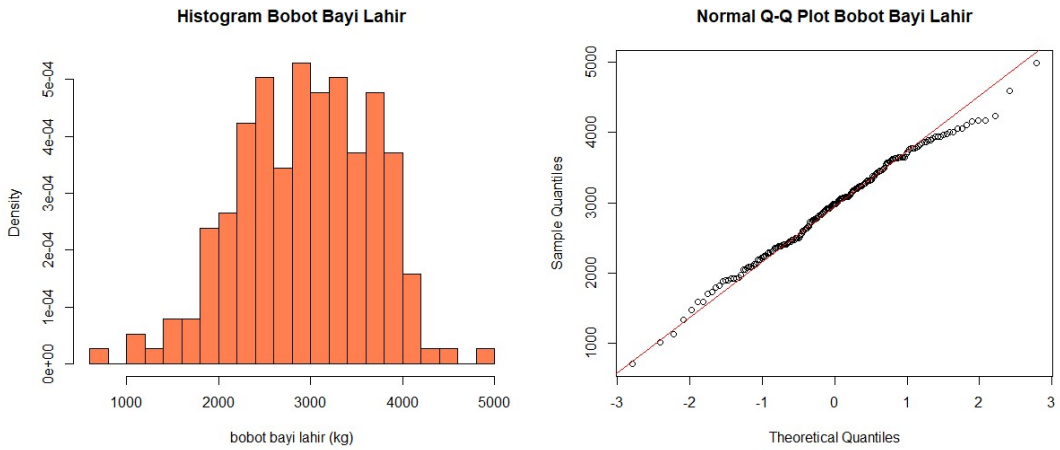


Gambar 1.21 Ilustrasi *normal Q-Q plot* dari data yang menjulur ke kanan dan ke kiri

Berikut ini adalah program R yang dapat digunakan untuk memeriksa apakah peubah bobot bayi lahir mengikuti sebaran normal menggunakan plot kuantil-kuantil normal. Fungsi yang dapat digunakan adalah `qqnorm()` untuk menampilkan *scatter plot* dari kedua kuantil dan `qqline()` untuk menampilkan garis lurus referensi untuk membantu melihat apakah titik-titik pada plot membentuk garis lurus yang diinginkan. Pada kasus data bobot ini terlihat bahwa secara umum titik-titik mengikuti garis lurus meskipun ada sedikit penyimpangan pada ujung kanan.

PROGRAM R 1.9

```
data.lahir <- read.csv("lowbwt.csv")
bobot <- data.lahir$bwt
hist(bobot, breaks=25, col="coral",
     xlab="bobot bayi lahir (kg)",
     main="Histogram Bobot Bayi Lahir",
     freq=FALSE)
qqnorm(bobot, main="Normal Q-Q Plot Bobot Bayi Lahir")
qqline(bobot, col = "red")
```



Gambar 1.22 Histogram dan *normal Q-Q plot* hasil program R 1.9

Selain melakukan pemeriksaan secara visual menggunakan Q-Q plot, terdapat juga uji formal untuk melakukan pemeriksaan sebaran, yang dikenal dengan sebutan *goodness of fit test*. Pada uji ini, hipotesis yang diuji adalah

- H0: data mengikuti sebaran hipotetik
- H1: data tidak mengikuti sebaran hipotetik

Tersedia berbagai macam uji formal di literatur untuk melakukan pengujian ini, namun pada tulisan ini hanya dibatasi mendiskusikan dua uji yaitu Chi-Square Test dan Kolmogorov-Smirnov Test. Keduanya dipilih karena menggunakan pendekatan yang berbeda. Chi-Square test, didasarkan pada perbandingan frekuensi amatan antara data contoh empirik dengan kondisi jika sebarannya mengikuti fungsi kepekatan/massa peluang tertentu. Sedangkan Kolmogorov-Smirnov test, didasarkan pada perbandingan antara fungsi sebaran kumulatif empirik dan fungsi sebaran kumulatif hipotetik.

Chi-Square Test

Uji ini barangkali termasuk uji kebaikan suai yang paling tua dan diusulkan oleh Karl Pearson. Secara sederhana, uji ini dapat dipandang seperti membandingkan histogram data dengan fungsi kepekatan/massa peluangnya. Uji Chi-Square menarik karena dapat diaplikasikan baik pada sebaran kontinu maupun diskret. Hanya saja pada saat mengimplementasikan uji ini, data harus di-kategorisasi (*binned*) seperti halnya kita membuat histogram. Dan tentu saja hasil tes akan bergantung pada proses kategorisasi ini. Kelemahan lain dari uji Chi-Square adalah bahwa uji ini memerlukan ukuran contoh yang besar agar pendekatan ke sebaran Chi-Square menjadi valid.

Hipotesis yang diuji pada uji ini adalah:

- H0: data mengikuti sebaran hipotetik
- H1: data tidak mengikuti sebaran hipotetik

Prosedur pengujian diawali dengan menyusun selang-selang nilai kategorisasi menjadi k kelas. Selanjutnya, untuk setiap kelas/selang nilai dihitung frekuensi amatan (*observed*) dan dinotasikan O_i. Menggunakan sebaran hipotetik yang akan dibandingkan, kita dapat menghitung peluang kejadian selang tersebut dan kemudian menghitung frekuensi harapannya (*expected*) yang dinotasikan E_i dan nilainya diperoleh dari $E_i = p_i \times n$.

Statistik uji yang digunakan adalah

$$\chi^2_{hitung} = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i}$$

dan χ^2_{hitung} mengikuti sebaran χ^2 dengan derajat bebas $(k - 1 - d)$ dengan d adalah banyaknya parameter yang diduga menggunakan data contoh.

Berikut ini adalah contoh program di R yang dapat digunakan untuk melakukan uji Chi-Square. Pertama, menggunakan fungsi `hist()` kita dapat melakukan pembuatan kelas-kelas sama lebar yang pada ilustrasi ini dibagi menjadi 25 kelas. Selanjutnya fungsi `pnorm()` digunakan untuk menghitung nilai peluang kumulatif normal yang nantinya akan diselisihkan menggunakan fungsi `rollapply()` sehingga diperoleh nilai peluang hipotetik.

Perhatikan pada saat menggunakan fungsi `pnorm()` kita perlu menyertakan nilai rata-rata dan simpangan baku sebaran yang dalam kasus ini diduga menggunakan rata-rata dan simpangan baku contoh.

PROGRAM R 1.10

```
data.lahir <- read.csv("lowbwt.csv")
bobot <- data.lahir$bwt
bb <- hist(bobot,breaks=25, right=FALSE)
p_kum <- pnorm(bb$breaks, mean=mean(bobot), sd=sd(bobot))

library(zoo)
p_norm <- rollapply(p_kum, 2, function(x) x[2]-x[1])
chisq.test(bb$counts, p=p_norm, rescale.p=TRUE,
           simulate.p.value=TRUE)
```

Program di atas akan menghasilkan *output* di bawah ini, dimana nilai statistik uji adalah 20.406 dan berdasarkan simulasi 200 kali diperoleh nilai-p (*p-value*) sebesar 0.4773 yang membawa kita untuk menerima hipotesis bahwa data mengikuti sebaran normal.

```
      Chi-squared test for given probabilities with
      simulated p-value (based on 2000 replicates)

data:  bb$counts
X-squared = 20.406, df = NA, p-value = 0.4773
```

Kolmogorov Smirnov Test

Seperti halnya uji Chi-Square, uji Kolmogorov-Smirnov ini digunakan untuk memeriksa apakah contoh acak yang kita miliki berasal dari sebuah sebaran yang diketahui fungsinya (sebaran hipotetik). Contoh acak tersebut yang merupakan $\{x_1, x_2, \ldots, x_n\}$ ditarik dari populasi tertentu yang nantinya akan dibandingkan dengan bentuk fungsi kumulatif $F^*(x)$ dengan suatu cara sehingga kita bisa mengatakan apakah cukup alasan untuk mengatakan bahwa $F^*(x)$ merupakan fungsi sebaran yang sesuai untuk data contoh yang kita miliki.

Salah satu cara yang logis untuk menyimpulkan hal tersebut adalah dengan membandingkan $F^*(x)$ terhadap fungsi sebaran kumulatif empiris $S(x)$ yang kita definisikan sebagai berikut. Andaikan $x_1, x_2, \ldots, x_n$ adalah contoh acak, suatu fungsi sebaran kumulatif empiris $S(x)$ adalah sebuah fungsi dari x yang menyatakan proporsi banyaknya x_i yang bernilai lebih kecil atau sama dengan x, untuk setiap x yang memenuhi $-\infty < x < \infty$ atau kita dapat tuliskan bahwa

$$S(x) = \frac{1}{n}\sum_{i=1}^{n} I_{\{x_i \le x\}}$$

dengan I adalah fungsi indikator yang bernilai 1 jika syarat yang tertuliskan terpenuhi dan 0 jika sebaliknya.

Uji Kolmogorov-Smirnov selanjutnya bekerja dengan membandingkan $F^*(x)$ terhadap $S(x)$. Statitik uji dari fungsi ini adalah D yang merupakan nilai terbesar (*supremum*) dari selisih antara $S(x)$ dan $F^*(x)$ atau dituliskan sebagai

$$D = \sup_x | F^*(x) - S(x) |$$

Tabel 1.1 merupakan tabel nilai kritis dari dari uji Kolmogorov-Smirnov yang digunakan untuk menyimpulkan Tolak H0 jika statistik uji D melebihi nilai kritis dalam tabel.

Berikut ini contoh program di R yang dapat digunakan untuk menguji menggunakan Uji Kolmogorov-Smirnov apakah data bobot badan bayi saat lahir mengikuti sebaran normal (dengan rata-rata dan simpangan baku menggunakan rata-rata dan simpangan baku contoh). Fungsi yang digunakan adalah `ks.test()` yang memerlukan setidaknya

dua vektor sebagai argumen yaitu vektor data dan vektor acuannya yang dalam hal ini adalah sebaran normal.

Tabel 1.1. Tabel nilai kritis uji Kolmogorov-Smirnov

	$1-\alpha$		
n	**0.9**	**0.95**	**0.99**
1	0.950	0.975	0.995
2	0.776	0.842	0.929
3	0.636	0.708	0.829
4	0.565	0.624	0.734
5	0.510	0.563	0.669
6	0.468	0.520	0.617
7	0.436	0.483	0.576
8	0.410	0.454	0.542
9	0.387	0.430	0.513
10	0.369	0.409	0.489
11	0.352	0.391	0.468
12	0.338	0.375	0.450
13	0.325	0.361	0.432
14	0.314	0.349	0.418
15	0.304	0.338	0.404
16	0.295	0.327	0.392
17	0.286	0.318	0.381
18	0.279	0.309	0.371
19	0.271	0.301	0.361
20	0.265	0.294	0.352

	$1-\alpha$		
n	**0.9**	**0.95**	**0.99**
21	0.259	0.287	0.344
22	0.253	0.281	0.337
23	0.247	0.275	0.330
24	0.242	0.269	0.323
25	0.238	0.264	0.317
26	0.233	0.259	0.311
27	0.229	0.254	0.305
28	0.225	0.250	0.300
29	0.221	0.246	0.295
30	0.218	0.242	0.290
31	0.214	0.238	0.285
32	0.211	0.234	0.281
33	0.208	0.231	0.277
34	0.205	0.227	0.273
35	0.202	0.224	0.269
> 35	$\frac{1.224}{\sqrt{n}}$	$\frac{1.358}{\sqrt{n}}$	$\frac{1.628}{\sqrt{n}}$

PROGRAM R 1.11

```
data.lahir <- read.csv("lowbwt.csv")
bobot <- data.lahir$bwt
ks.test(bobot, "pnorm", mean(bobot), sd(bobot))
```

Output yang diperoleh dari program di atas adalah yang menghasilkan statistik uji D sebesar 0.043484 dan nilai p sebesar 0.8762. Berdasarkan ini kita tidak menolak hipotesis bahwa distribusinya mengikuti sebaran normal.

```
      One-sample Kolmogorov-Smirnov test
data:  bobot
D = 0.043484, p-value = 0.8672
alternative hypothesis: two-sided
```