1. Eksplorasi Sebaran Data

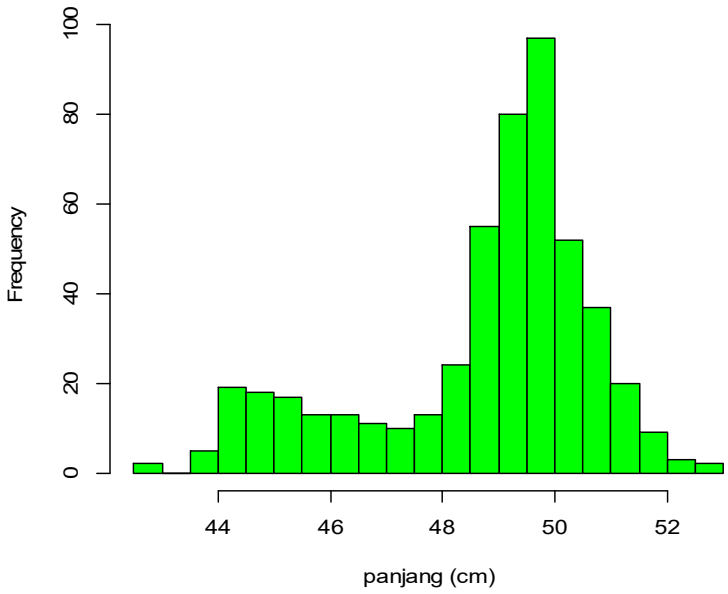
1.1. Pengantar

Sebaran data merupakan informasi penting dalam proses eksplorasi karena menyangkut karakteristik probabilistik dari data yang kita miliki. Perlu dipahami bahwa informasi mengenai sebaran data ini terkait dengan bentuk sebaran dari populasi asal dari sampel data yang kita miliki yang dalam analisis statistika lanjut nantinya banyak berhubungan dengan metode analisis terutama proses inferensia dari statistik yang diperoleh.

1.2. Histogram

Dari akar katanya, histogram adalah gabungan dua kata dasar yaitu Histos (yang berarti sesuatu yang diatur tegak) dan Gramma (yang berarti gambar atau tulisan). Dengan demikian, secara sederhana kita dapat menyatakan bahwa histogram adalah suatu grafik yang menggambarkan distribusi dari data (kontinu) yang berupa deretan batang sama lebar berdampingan yang tingginya menggambarkan banyaknya data untuk berbagai selang nilai.

Gambar berikut merupakan contoh salah satu bentuk dari histogram yang dihasilkan dari xxx amatan untuk peubah panjang bayi saat lahir (dalam cm).



Sumbu horizontal dari suatu histogram menampilkan selang-selang nilai variabel yang akan dilihat distribusinya, dalam hal ini adalah panjang bayi lahir dengan selang masing-masing selebar 0.5 cm. Sementara itu, sumbu vertikal histogram menunjukkan

frekuensi atau banyaknya amatan dalam gugus data dari setiap selang nilai. Kadangkala, nilai frekuensi digantikan dengan nilai persentase yaitu nilai frekuensi dibagi dengan ukuran gugus data atau ukuran contoh. Setiap selang nilai kemudian dilambangkan dalam bentuk batang-batang dengan tinggi yang sesuai dengan frekuensinya. Karena peubah yang digambarkan bersifat kontinu, maka antara selang nilai yang bersebelahan digambarkan dalam bentuk batang tanpa celah. Sekali lagi, tinggi rendahnya batang menggambarkan besar kecilnya frekuensi masing-masing selang nilai. Pada contoh di atas misalnya, batang yang paling tinggi adalah sekitar 49 dan 50 cm yang berarti bahwa panjang bayi yang paling banyak dijumpai adalah pada rentang nilai tersebut.

Berdasarkan definisi yang dinyatakan seperti di atas, maka kita dapat memperoleh histogram melalui tahapan pembuatan histogram adalah sebagai berikut:

1. susun selang-selang nilai yang sama lebar, dan meliputi seluruh nilai data yang dimiliki
2. hitung banyaknya amatan yang tercakup dalam masing-masing selang
3. pada sumbu mendatar, tandai untuk setiap batas selang nilai
4. pada setiap selang nilai, gambarkan batang yang tingginya sesuai dengan frekuensinya

Perangkat lunak R menyediakan berbagai fungsi untuk menghasilkan histogram. Salah satu fungsi dasar yang dapat digunakan adalah **hist** yang ada pada package **graphics**. Berikut ini adalah ilustrasi program menghasilkan histogram di R.

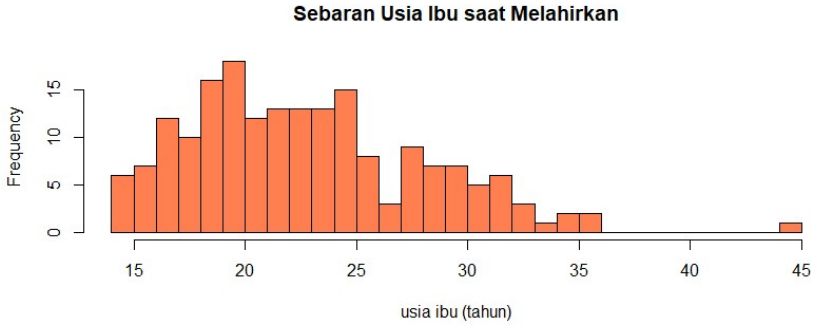
```
data.lahir <- read.csv("D:/lowbwt.csv")
colnames(data.lahir)
hist(data.lahir$age, breaks=25, col="coral",
     xlab="usia ibu (tahun)",
     main="Sebaran Usia Ibu saat Melahirkan")
```

Program di atas diawali dengan membaca file data lowbwt.csv menggunakan fungsi read.csv() yang dilanjutkan dengan melihat nama-nama kolom dari data frame data.lahir menggunakan fungsi colnames(). Gugus data ini adalah gugus data yang digunakan pada Hosmer dan Lemeshow (2013) tentang berat badan bayi lahir rendah yang berisi 189 amatan. Salah satu kolom yang ada pada data frame tersebut adalah age dan fungsi hist() selanjutnya digunakan untuk menampilkan gambar histogram. Opsi-opsi dari fungsi yang digunakan pada ilustrasi program di atas adalah:

- breaks, opsi untuk menentukan banyaknya selang/batang yang digambar
- col, opsi untuk menentukan warna dari batang
- xlab, opsi untuk menentukan label dari sumbu X

- main, opsi untuk menentukan judul gambar

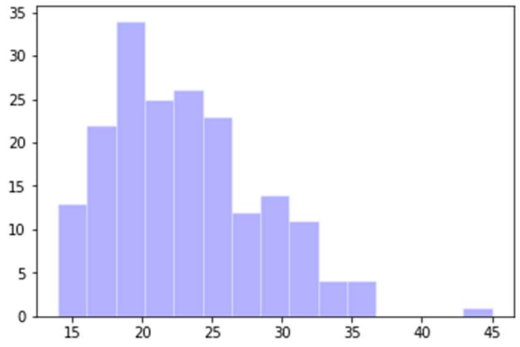
Hasil dari program di atas adalah Gambar di bawah ini



Sementara itu, berikut ini adalah program yang dapat digunakan untuk menghasilkan histogram dari peubah age (usia ibu saat melahirkan) menggunakan bahasa Python. Fungsi yang digunakan adalah hist yang ada pada modul matplotlib.pyplot.

```
import pandas as pd
data_lahir = pd.read_csv("D:/lowbwt.csv")
import matplotlib.pyplot as plt
plt.hist(data_lahir['age'], 15,  edgecolor='white',
        facecolor='blue', alpha=0.3)
plt.show()
```

Pada perintah di atas digunakan beberapa opsi seperti edgecolor untuk menentukan warna sisi batang dan facecolor untuk menentukan warna batang.

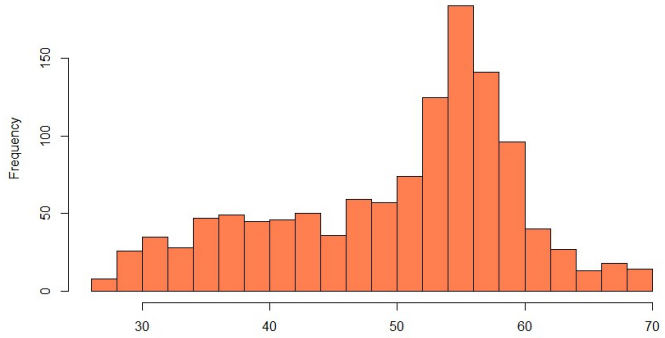


Histogram yang diperoleh dari gugus data suatu peubah tertentu memiliki banyak kegunaan. Beberapa kegunaan dari histogram akan didiskusikan pada beberapa paragraf di bawah ini.

Pertama, histogram dapat memberikan informasi ukuran pemusatan dan penyebaran data secara cepat, meskipun ukuran contohnya sangat besar. Ukuran pemusatan dapat segera dikenali dari histogram dengan cara mengidentifikasi selang(-selang) nilai mana

yang memiliki batang paling tinggi. Seperti yang telah dijelaskan bahwa tinggi rendahnya batang menyatakan frekuensi selang nilai tersebut pada data, sehingga beberapa selang berdekatan dengan batang paling tinggi mengindikasikan bahwa sebagian besar amatan ada disana dan nilai itulah yang disebut sebagai ukuran pemusatan.

Sebagai ilustrasi, berikut ini adalah histogram dari data usia karyawan suatu perusahaan. Dari sumbu horizontalnya kita dapat membaca bahwa range nilai usia karyawan di perusahaan tersebut antara 25 tahun hingga 70 tahun. Selang nilai usia dengan batang paling tinggi adalah pada selang 54 – 56 tahun dan 56 – 58 tahun. Dengan kata lain kita dapat mengatakan bahwa disanalah nilai pemusatan dari data usia. Sebagian besar karyawan memiliki usia pada rentang 52 hingga 60 tahun tersebut.



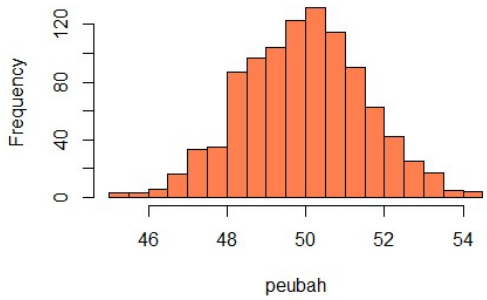
Kegunaan kedua dari histogram adalah dapat mengenali pola umum dari sebaran data yang kita miliki. Dari gugus data yang ada, kita dapat memperoleh bentuk sebaran tertentu yang bisa jadi berbeda dengan gugus data lainnya. Gambar berikut memberikan bentuk-bentuk tipikal dari histogram yang banyak kita temui.

Bentuk sebaran pertama yang banyak didapat dari sebuah gugus data adalah bentuk pola simetrik yang menyerupai sebaran normal (Gambar (a)). Histogram ini memiliki satu buah puncak yang berada di tengah dengan ekor pada bagian kiri dan kanan relatif seimbang. Pola simetris normal seperti ini banyak diperoleh dari data-data yang berasal dari pengukuran morfologis seperti tinggi badan dan berat badan manusia. Dengan pola simetris ini, nilai rata-rata dari data cenderung sama dengan dengan nilai mediannya.

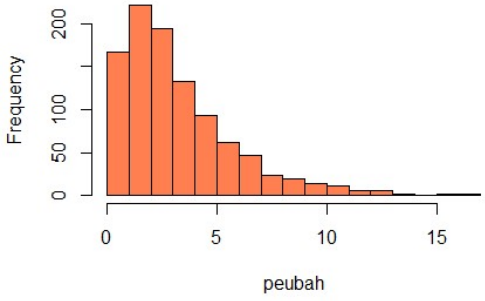
Pola atau bentuk tipikal sebaran berikutnya adalah yang disajikan pada Gambar (b). Sebaran ini memiliki ukuran pemusatan yang letaknya cenderung di bagian kiri dari rentang data keseluruhan dan memiliki ekor yang cenderung lebih panjang di sebelah kanan. Istilah yang sering digunakan untuk bentuk ini adalah menjulur ke kanan (skew-to-the-right distribution, right-tailed distribution). Bentuk ini menandakan adanya amatan-amatan pada gugus data yang nilainya jauh lebih besar dari amatan-amatan pada umumnya, meskipun frekuensinya tidak sangat banyak. Karenanya, gugus data dengan bentuk sebaran seperti ini akan memiliki nilai rata-rata yang lebih besar dibandingkan

mediannya. Bentuk sebaran menjulur ke kanan banyak dijumpai pada peubah-peubah yang menggambarkan tingkat kondisi ekonomi, misalnya nilai saldo tabungan nasabah, nilai pendapatan rumah tangga penduduk suatu negara, dan sebagainya. Peubah lain yang memiliki bentuk sebaran menjulur ke kanan adalah nilai ujian dari mata kuliah yang sulit, sehingga sebagian besar mahasiswa memiliki nilai yang tidak besar tapi ada satu dua mahasiswa yang nilainya sangat baik.

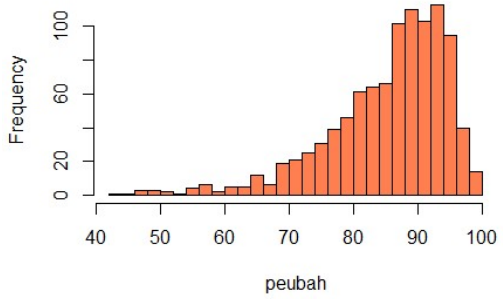
Sebaliknya, ada juga bentuk sebaran yang disebut menjulur ke kiri (skew-to-the-left distribution, left-tailed distribution) dimana sebagian besar amatan memiliki nilai yang cenderung besar namun ada sebagian kecil amatan yang nilainya sangat kecil sehingga digambarkan pada batang pendek di bagian kiri histogram. Peubah nilai ujian mahasiswa pada mata kuliah mudah biasanya akan seperti ini polanya, yang tampilan visualnya seperti pada Gambar (c). Gugus data dengan sebaran menjulur ke kiri memiliki nilai rata-rata yang lebih rendah dibandingkan nilai mediannya.



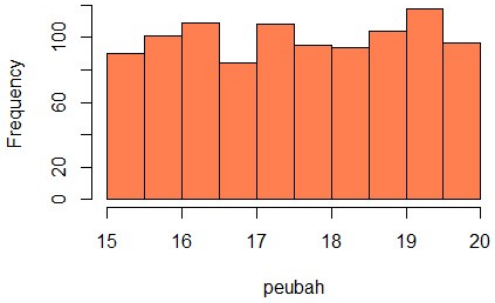
(a). histogram dengan pola simetrik



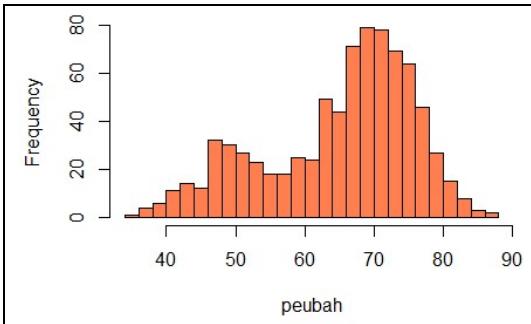
(b). histogram dengan pola menjulur ke kanan



c. histogram dengan pola menjulur ke kiri



d. histogram dengan sebaran seragam

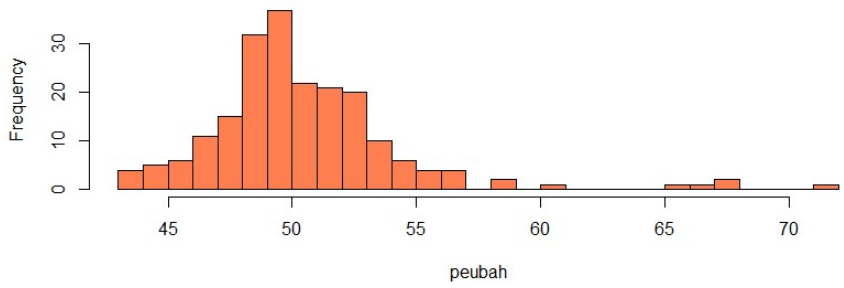


e. histogram yang berpola bimodal

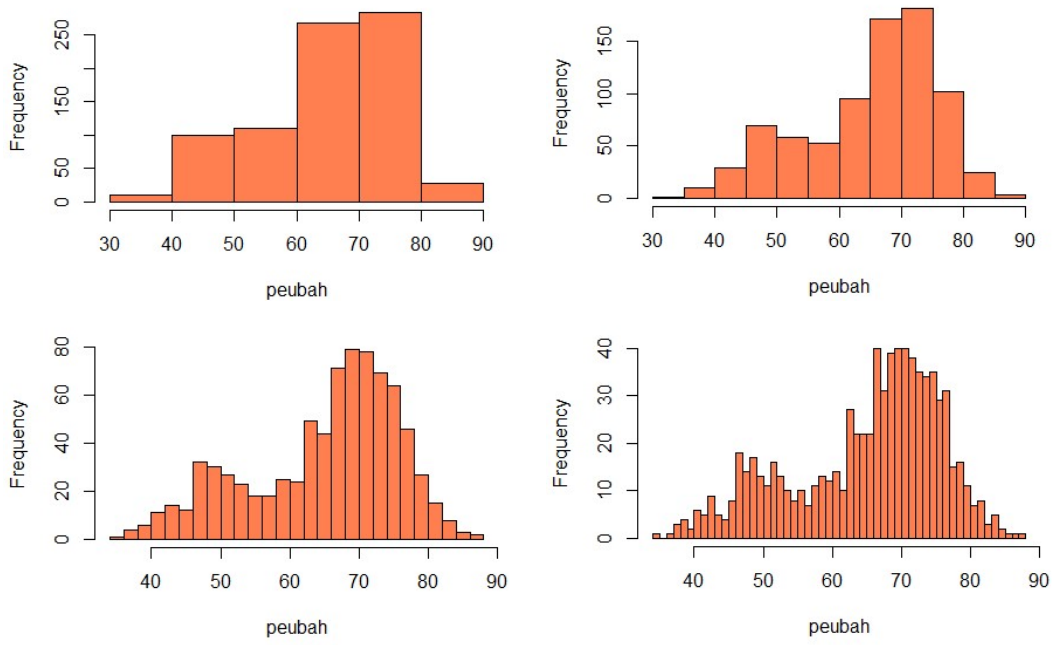
Bentuk sebaran lain yang mungkin adalah sebaran batang yang tingginya seragam antara satu batang dengan batang yang lain. Histogram dengan bentuk seragam seperti pada Gambar (d) mengindikasikan ketiadaan selang nilai yang dominan. Bentuk ini misalnya dapat dijumpai pada data usia siswa di suatu SMA.

Pola lain terkait dengan bentuk sebaran yang dapat dikenali menggunakan histogram adalah adanya puncak pada lebih dari satu titik. Gambar (e) menampilkan ilustrasi histogram dengan dua puncak yang sering disebut sebagai bimodal (memiliki dua modus). Bentuk semacam ini dapat terjadi karena pada data tercampur dua subpopulasi yang karakteristiknya berlainan. Misalnya saja, dari data diameter tanaman suatu perkebunan sawit akan terbentuk histogram dengan beberapa puncak karena tanaman tersebut memiliki umur tanam yang bervariasi dari satu blok-kebun ke blok-kebun yang lain.

Kegunaan ketiga dari histogram yang dapat disebutkan pada tulisan ini adalah bahwa histogram dapat membantu mengidentifikasi dengan cepat keberadaan amatan yang nilainya ekstrim dan berbeda dengan kebanyakan amatan lainnya. Amatan dengan nilai ekstrim ini bisa jadi memang situasi yang sebenarnya, namun bisa juga terjadi karena ada kesalahan pencatatan atau perbedaan satuan. Jika ada amatan dengan kondisi seperti ini maka pada histogram akan terdapat batang yang letaknya jauh terpisah dari batan-batang yang lain seperti disajikan pada Gambar berikut.



Perlu dipahami bahwa dalam pembuatan histogram, analis memiliki kebebasan dalam menentukan banyaknya batang yang digunakan sebagai representasi selang-selang nilai pada gugus data yang tersedia. Menggunakan gugus data yang sama, berikut ini beberapa bentuk histogram yang didapatkan dengan menentukan banyaknya selang yang berbeda-beda mulai dari banyaknya selang yang sedikit (hanya 6 batang) hingga selang yang banyak jumlahnya (sekitar 45 batang).



Tampak bahwa pada histogram yang pertama, dimana selang yang digunakan terlalu sedikit, keberadaan puncak pada bagian kiri histogram tidak terlalu kelihatan sebagaimana terlihat pada histogram kedua dan ketiga. Sementara pada histogram terakhir, yang memiliki selang sangat banyak, terdapat bentukan batang-batang bergerigi yang menjorok ke atas. Dalam istilah lain, histogram keempat ini memiliki batang yang berbentuk *spiky* yaitu pola seperti tanduk-tanduk yang justru mengganggu bentuk umum dari tampilan sebaran yang sesungguhnya.

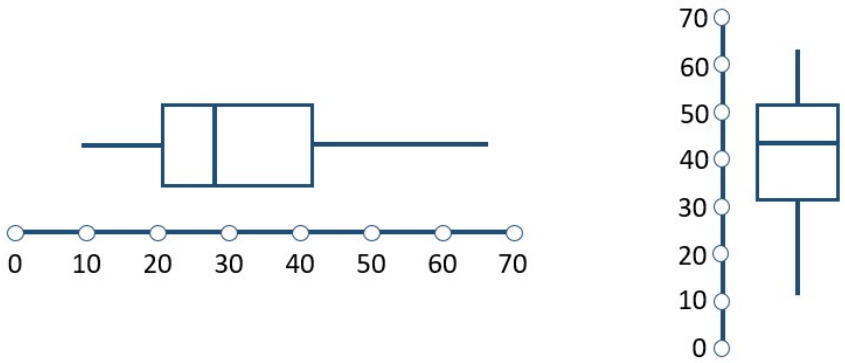
Karena hal tersebut di atas, terdapat beberapa usulan formula dalam penentuan banyaknya selang atau batang dalam pembuatan histogram. Banyaknya selang ini selanjutnya tentu saja menentukan lebar selang yang digunakan. Semakin banyak batang yang digunakan, maka selang-selang nilai yang dibentuk akan semakin sempit. Formula penentuan banyaknya selang optimum yang bisa ditemukan di literatur adalah sebagai berikut:

- Akar kuadrat dari banyaknya amatan
  $k = \sqrt{n}$

- Formula yang diusulkan H.A. Sturges
  $k = \lceil \log_2 n + 1 \rceil$
- Formula yang diusulkan Rice University
  $k = \left\lceil 2n^{1/3} \right\rceil$
- Formula yang diusulkan DP Doane
  $k = \frac{3.5s}{n^{1/3}}$
- Formula yang diusulkan David Freedman dan P Diaconis
  $k = \frac{2\ \mathrm{IQR}}{n^{1/3}}$

1.3. Boxplot

Teknik grafik lain yang juga dapat digunakan untuk melihat bentuk sebaran data adalah boxplot atau diagram kotak garis. Sesuai dengan namanya, diagram ini berbentuk sebuah kotak atau persegi panjang horizontal yang di kanan-kirinya ditambahkan garis mendatar. Tentu saja tampilan ini dapat diubah menjadi kotak vertikal dengan garis tegak pada bagian atas dan bawah kotak seperti yang ada pada Gambar berikut.



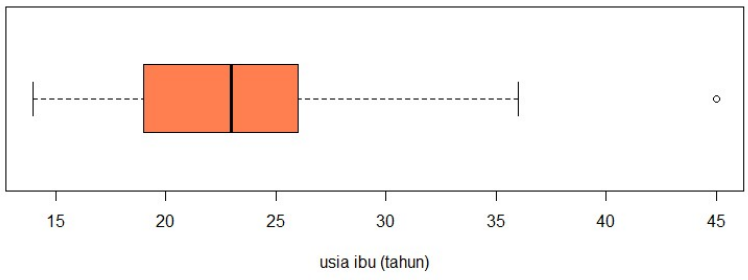
Prosedur untuk memperoleh suatu diagram kotak garis horizontal adalah sebagai berikut:

- Hitung beberapa statistik, meliputi:
  - statistik lima serangkai (Min, Q1, Q2, Q3, Max)
  - batas atas BA = Q3 + 3/2 (Q3-Q1)
  - batas bawah BB = Q1 - 3/2 (Q3-Q1)
- deteksi keberadaan pencilan, yaitu data yang nilainya kurang dari BB atau data yang lebih besar dari BA
- gambar kotak horizontal, dengan batas kiri Q1 sampai batas kanan Q3, dan letakkan tanda garis di tengah kotak pada posisi Q2

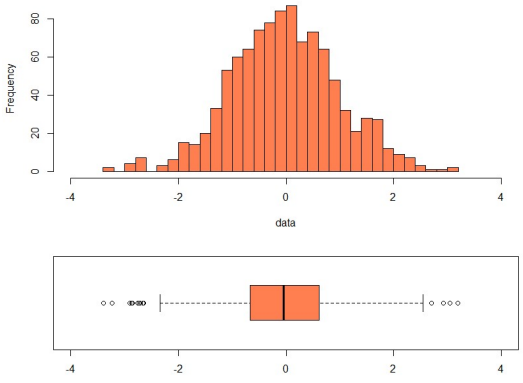
- Tarik garis ke kanan, mulai dari Q3 sampai data terbesar di dalam batas atas
- Tarik garis ke kiri, mulai dari Q1 sampai data terkecil di dalam batas bawah
- Tandai pencilan dengan lingkaran kecil

Program R sederhana untuk menghasilkan boxplot dari data yang sama dengan ilustrasi histogram sebelumnya adalah sebagai berikut. Fungsi yang dapat digunakan untuk menghasilkan boxplot horizontal adalah fungsi **boxplot()** dengan menambahkan opsi **horizontal=TRUE**. Keberadaan amatan yang berupa pencilan akan ditandai dengan titik dalam bentuk lingkaran kecil.

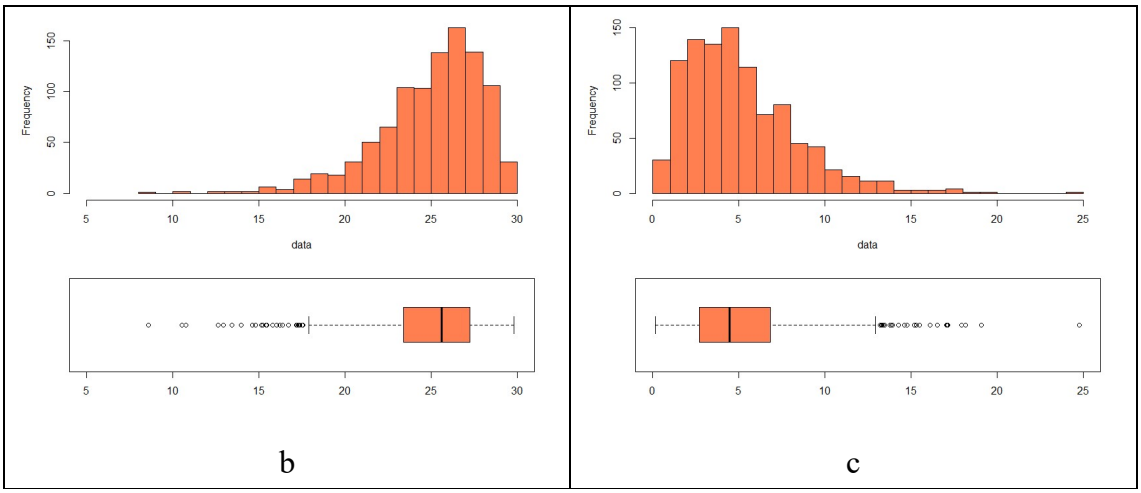
```
boxplot(data.lahir$age, horizontal=TRUE,
        col="coral", xlab="usia ibu (tahun)")
```



Berikut ini adalah bentuk-bentuk diagram kotak garis yang dipadankan dengan bentuk histogram dari data. Pada gugus data dengan sebaran berbentuk simetris, diagram kotak garisnya juga terlihat simetris baik pada bagian kotak maupun bagian garisnya. Pada bagian kotak akan terbagi dua dengan lebar yang hampir sama. Demikian juga untuk bagian garis yang terlihat sama panjang antara yang ke kiri dan ke kanan.



a



Sementara itu, pada data dengan sebaran menjulur ke kiri (Gambar (b)) nampak bahwa bagian gambar kotak dan garisnya akan cenderung lebih panjang pada bagian kiri dibandingkan kotak dan garis yang di bagian kanan. Sebaliknya untuk data dengan sebaran menjulur ke kanan, maka bagian kanan kotak dan garis yang lebih panjang.

1.4. Dugaan Bentuk Fungsi Kepekatan

Histogram dan boxplot, seperti yang telah didiskusikan sebelumnya, dapat dijadian alat untuk mengenali bentuk sebaran dari data yang kita miliki. Pada situasi tertentu kita ingin memperoleh dugaan fungsi sebaran berupa fungsi kepekatan peluang dari data. Fungs kepekatan merupakan fungsi yang menyatakan struktur peluang dari setiap kemungkinan nilai peubah. Jika kita memiliki fungsi kepekatan dari suatu peubah maka kita dapat menentukan besarnya peluang untuk sembarang selang nilai yang kita inginkan.

Pengetahuan terhadap fungsi kepekatan ini memiliki manfaat pada beberapa hal. Pertama, kita dapat mengenali dengan baik sebaran data jika kita mengetahui fungsi kepekatannya. Seperti yang telah dibicarakan di atas, dengan fungsi kepekatan ini kita dapat membangkitkan nilai peluang untuk berbagai selang nilai. Informasi peluang ini penting dalam kaitannya dengan eksplorasi data secara umum.

Manfaat lain dari pengetahuan terhadap fungsi kepekatan adalah bahwa fungsi ini akan banyak membantu dalam proses analisis dengan teknik simulasi. Berbagai teknik simulasi statistika memerlukan pembangkitan bilangan acak dengan sebaran yang spesifik sesuai dengan karakteristik populasi yang diinginkan. Pengetahuan mengenai fungsi kepekatan ini akan membantu simulasi memperoleh data yang lebih sesuai yang pada ujungnya akan memberikan kesimpulan hasil simulasi secara sah dan memuaskan.

Penduga Naïve

Pendekatan pertama untuk menduga fungsi kepekatan peluang berdasarkan data contoh yang kita miliki adalah yang disebut penduga naïve. Dengan pendekatan ini, penduga bagi fungsi kepekatan dari suatu peubah X adalah

$$\hat{f}_h(x) = \frac{1}{2hn}(\text{banyaknya data pada selang } (x-h, x+h))$$

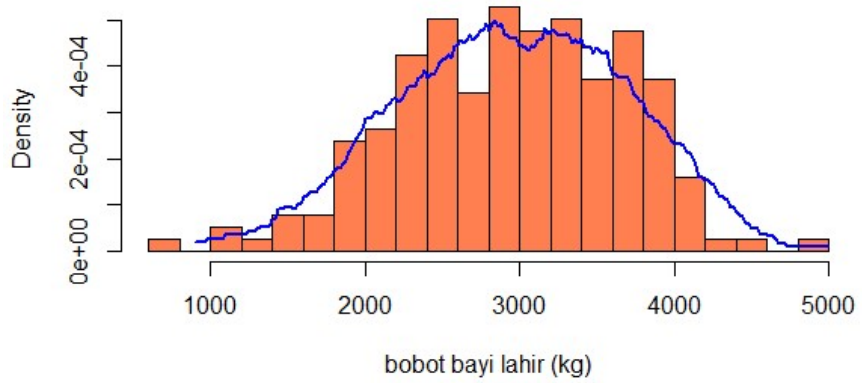
dengan n adalah ukuran contoh dan h adalah suatu konstanta yang merupakan setengah dari lebar selang.

Berikut ini adalah program R untuk menghasilkan penduga naïve bagi sebaran data peubah bwt (bobot bayi saat lahir) yang ada pada data lowbwt yang sudah disinggung sebelumnya. Selain menghitung nilai dugaan fungsi kepekatan f(x), program ini juga menggambarkan secara tumpang tindih histogram data dan fungsi kepekatannya.

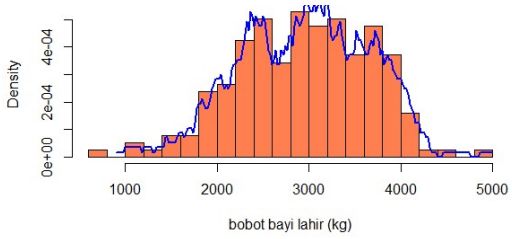
```
data.lahir <- read.csv("D:/lowbwt.csv")
bobot <- data.lahir$bwt
hist(bobot, breaks=25, col="coral",
     xlab="bobot bayi lahir (kg)",
     main="",
     freq=FALSE)


h = 500
n = length(bobot)
x <- seq(900, 5000, by=20)
fx <- NULL
for (i in 1:length(x)){
  fx[i] = sum(ifelse(abs(bobot - x[i]) < h , 1, 0)) /
(2*h*n)
}
lines(x, fx, type="l", col="blue", lwd=2)
```

Program menghitung nilai dugaan f(x) dapat dijelaskan sebagai berikut. Perintah sum(ifelse(abs(bobot - x[i]) < h , 1, 0)) digunakan untuk menghitung banyaknya amatan yang berada pada selang (x – h, x + h) dengan terlebih dahulu mengkonversi amatan yang bernilai dalam selang tersebut menjadi 1 dan yang di luar selang dikonversi menjadi 0. Baris lengkap dari perintah fx[i] adalah menyimpan nilai fungsi kepekatan untuk setiap nilai x pada selang antara 900 hingga 5000 kg. Karena tidak mungkin kita menghitung f(x) pada sembarang nilai, pada program ini hanya nilai bobot atau x antara 900 hingga 5000 dengan interval 20 kg saja yang dihitung. Semakin kecil interval itu akan membuat komputasi semakin lama karena semakin banyak nilai x yang diduga nilai f(x)-nya. Dalam program ini besaran h yang digunakan adalah 500, dan grafik

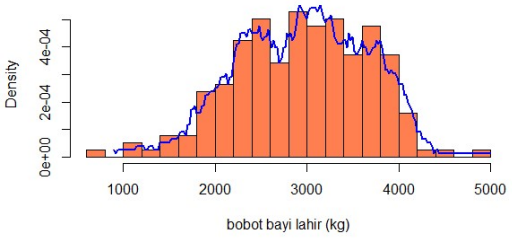
yang diperoleh dari program tersebut adalah sebagai berikut dimana kuva biru merupakan kurva f(x).



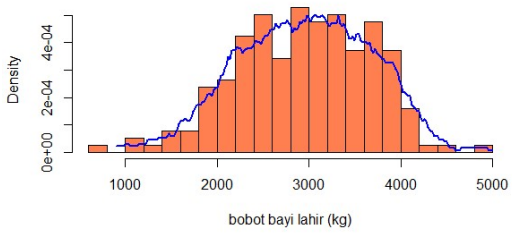
Pemilihan nilai h (lebar jendela) akan mempengaruhi bentuk kurva dari dugaan f(x). Kurva f(x) yang diperoleh dari nilai h yang besar akan cenderung menghasilkan kurva yang sangat mulus, sedangkan nilai h yang kecil akan menghasilkan kurva yang bergerigi (spiky). Untuk mempermudah menjelaskan fenomeda ini, berikut ini disajikan ilustrasi kurva f(x) pada kasus yang sama dengan yang diberikan sebelumnya pada data bobot lahir bayi dengan menggunakan beberapa nilai h yang berbeda.



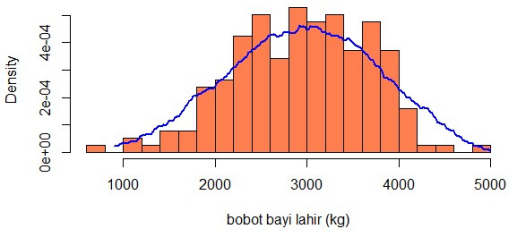
h = 150



h = 200



h = 350



h = 750

Formulasi penduga naïve dari fungsi kepekatan yang telah disampaikan di atas dapat pula ditulis ulang dalam bentuk

$$\hat{f}_h(x) = \frac{1}{hn}\sum_{i=1}^{n} w\left(\frac{x - x_i}{h}\right)$$

dengan xi adalah nilai data pada amatan ke-i, untuk i = 1, 2, …, n, sedangkan w() adalah fungsi yang didefinisikan sebagai

$$w(z) = \begin{cases} 1/2 & \text{jika } |z| < 1 \\ 0 & \text{selainnya} \end{cases}$$

Bentuk di atas akan membantu untuk menuliskan bentuk umum penduga berikutnya yang disebut sebagai penduga kernel.

Penduga Kernel

Penduga kernel merupakan proses pemulusan dari penduga fungsi kepekatan berdasarkan data contoh yang diperoleh menggunakan formula

$$\hat{f}_h(x) = \frac{1}{hn}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

Jika pada penduga naïve, nilai bobot w() diberikan sama kepada setiap nilai yang berada dalam selang (x-h, x+h) yaitu sebesar 1/2, pada penduga ini bobot diberikan sebesar K() yang merupakan fungsi kernel dan nilainya akan secara umum mengecil jika amatannya semakin jauh dari titik pusat selang yaitu x. Fungsi K() harus memenuhi syarat sebagai fungsi yang bersifat non negatif dan integralnya sama dengan 1 (satu).

Beberapa fungsi kernel yang biasa digunakan adalah

1. Uniform Kernel

   $$\mathrm{K(t)} = \frac{1}{2}\mathrm{I}(|\,\mathrm{t}\,| \le 1)$$

2. Gaussian Kernel

   $$K(t) = \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}}$$

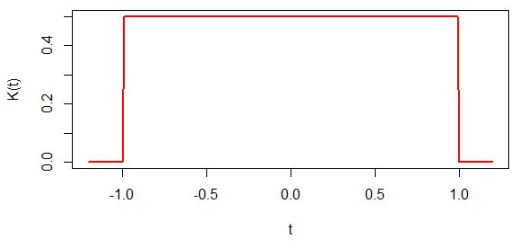
3. Epanechnikov Kernel

   $$K(t) = \begin{cases} \frac{3}{4}(1 - t^2), & |t| < 1 \\ 0 & \text{otherwise} \end{cases}$$

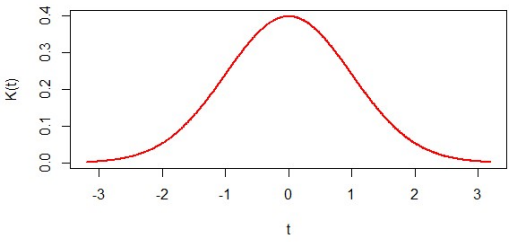
4. Biweight Kernel

$$K(t)=\begin{cases}\frac{15}{16}(1-t^2)^2, & |t|<1\\ 0 & \text{otherwise}\end{cases}$$

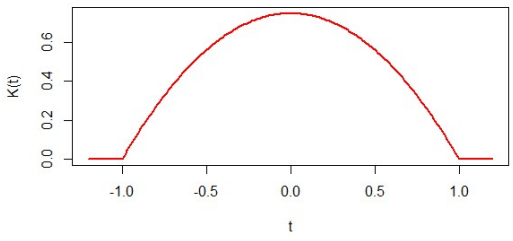
Berikut ini adalah kurva dari keempat fungsi kernel di atas.



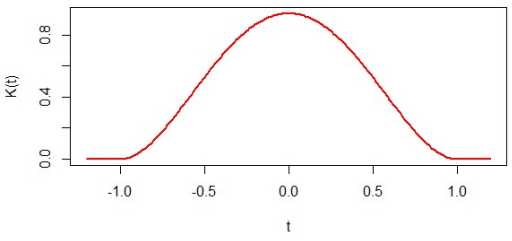
a. uniform



b. Gaussian

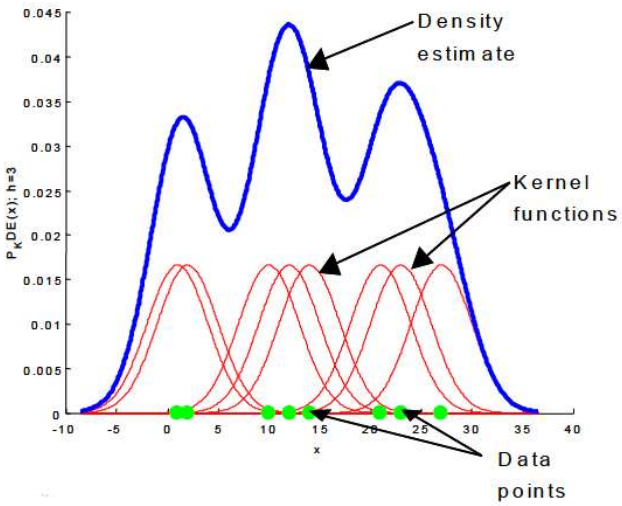


c. Epanechnikov



d. Biweight

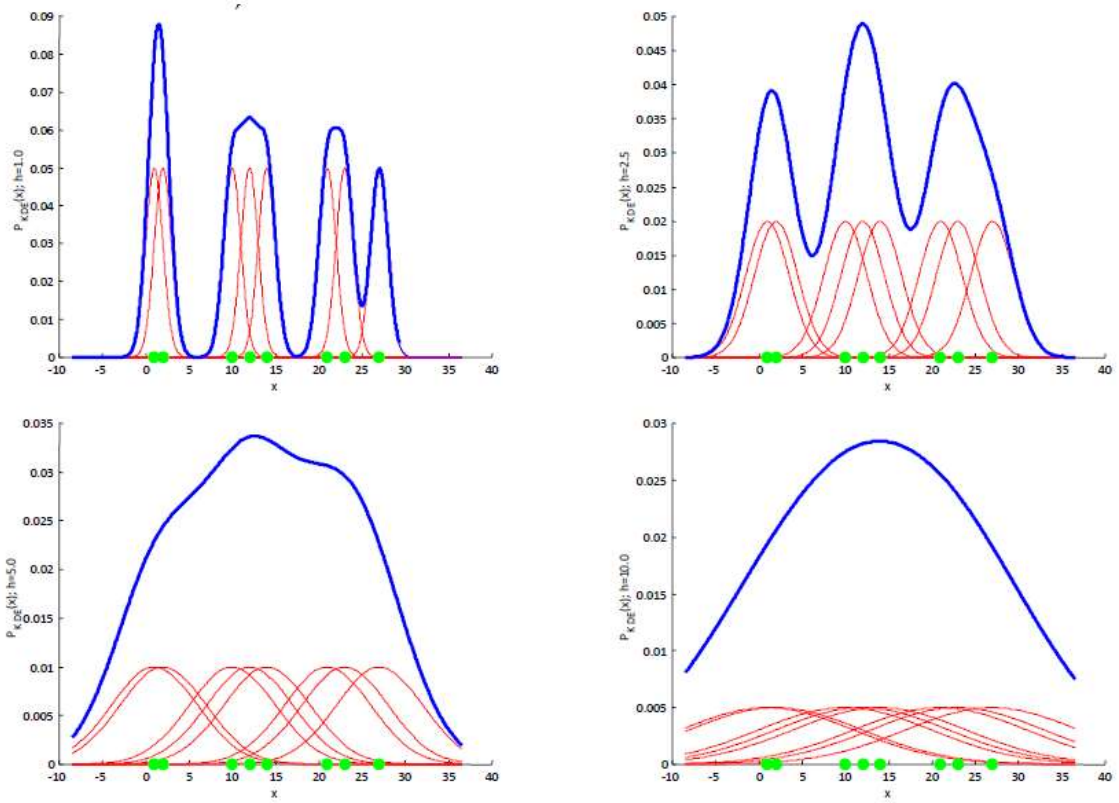
Pada prinsipnya, penduga kernel untuk fungsi kepekatan tidak lain adalah jumlah dari fungsi kernel. Ilustrai berikut menggambarkan bagaimana pendug kernel bekerja. Di sekitar setiap titik amatan (titik berwarna hijau) dapat diperoleh kurva fungsi kernel. Jika ada beberapa titik amatan yang nilai berdekatan maka akan ada kurva fungsi kernel yang saling berdekatan pula sehingga pada saat dijumlahkan di sekitar nilai tersebut akan memiliki nilai fungsi kepekatan yang lebih tinggi dibandingkan titik-titik lain yang amatannya sedikit.



Selanjutnya perlu dipahami bahwa bentuk kurva penduga fungsi kepekatan akan sangat tergantung pada lebar jendela (bandwidth) h. Jika lebar jendela h kecil maka kurva fungsi kernel akan cenderung lebih kurus dan memberi nilai 0 pada banyak nilai x. Karenanya maka hasil penjumlahan fungsi kernel akan berbentuk lebih bergerigi (spiky).

Sementara itu nilai lebar jendela yang besar akan membuat kurva fungsi kernel menjadi melebar dan tidak banyak yang bernilai nol, sehingga ketika dijumlahkan akan memperoleh penduga fungsi kepekatan yang cenderung datar dan tidak banyak lembah.

Gambar di bawah ini menyajikan lustrasi bagaimana bentuk dari fungsi kernel akan berubah-ubah jika lebar jendela diubah dan ini pada akhirnya akan mempengaruhi bentuk dari penduga fungsi kepekatan peluang.



Dari ilustrasi ini jelas bahwa jika lebar jendela dibuat terlalu kecil maka kurva fungsi kepekatan akan cenderung banyak lembah dan puncak, sedangkan kalau terlalu besar akan menghasilkan kurva yang semakin mulus. Karenanya ada juga yang menyebut lebar jendela ini sebagai parameter pemulusan (smoothing parameter) pada pendekatan kernel.

Pada program R fungsi yang dapat digunakan untuk menghasilkan penduga fungsi kepekatan menggunakan metode kernel adalah fungsi density(). Berikut ini adalah contoh program yang digunakan untuk mendapatkan penduga fungsi kepekatan menggunakan kernel Gaussian, dan selanjutnya mengoverlay grafik kurva fungsi kepekatan dan histogram data.

```
kepekatan <- density(data.lahir$age, bw=1,
                     kernel="epanechnikov")
hist(data.lahir$age, freq=FALSE, breaks=15,
     col="skyblue1", main="",
     xlab="usia ibu (tahun)")
lines(kepekatan, col="blue", lwd=2,
      main="", ylim=c(0, 0.09))
```

Opsi utama yang ada pada fungsi density() adalah bw dan kernal. Opsi bw digunakan untuk menentukan lebar jendela yang digunakan sedangkan opsi kernel digunakan untuk memilih fungsi kernel apa yang digunakan. Fungsi density() ini menyediakan beberapa pilihan fungsi kernel yaitu: "gaussian", "epanechnikov", "rectangular", "triangular", dan "biweight".

1.5. Perbandingan Bentuk Sebaran Beberapa Populasi

Jika kita memiliki data yang berasal dari beberapa (sub)populasi, sering menjadi menarik untuk membandingkan karakteristik dari satu populasi dengan populasi lainnya. Salah satu yang bisa kita bandingkan adalah dengan melihat perbandingan bentuk sebarannya.

Untuk membandingkan bentuk sebaran, tidak mudah melalukannya secara visual menggunakan histogram. Beberapa histogram bisa saja kita buat, namun jika ditumpang-tindihkan dalam satu buah gambar akan terlihat berdesakan dan sulit menangkap informasinya. Cara visualisasi yang bisa digunakan adalah dengan membandingkan bentuk dari fungsi kepekatannya. Karena visualnya hanya berupa kurva, maka jika ada beberapa kurva dalam satu gambar masih mudah bagi kita untuk membaca dan memperoleh informasi dari gambar tersebut.

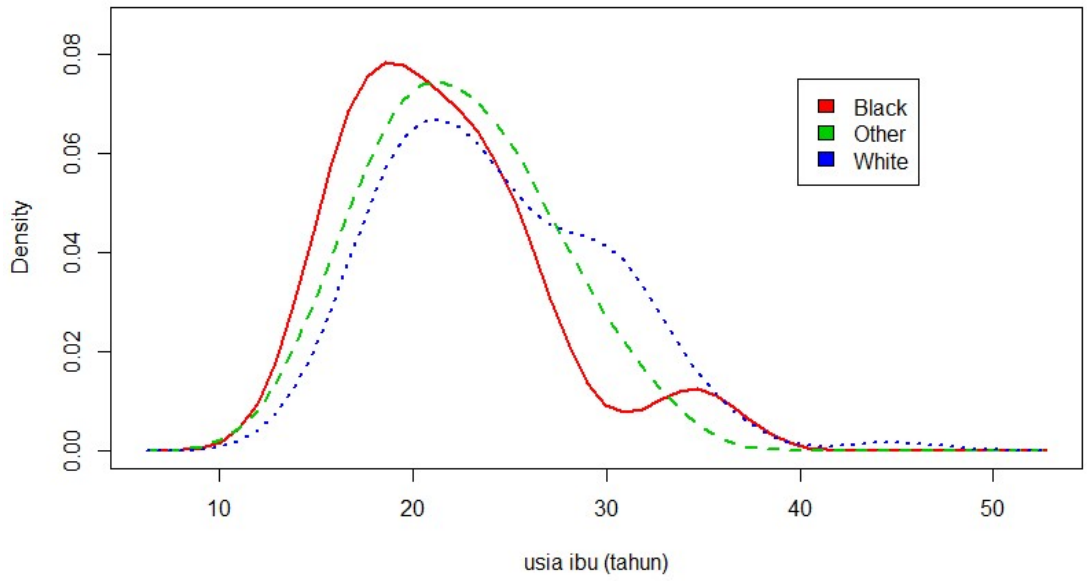
Fungsi density() pada R yang sudah kita gunakan sebelumnya dapat menjadi pilihan. Dengan memisahkan gugus data asal menjadi beberapa subset sesuai dengan kelompok yang ada, kita dapat menjalankan fungsi density() pada masing-masing kelompok. Namun pada R juga tersediap package lain yaitu sm yang menyediakan fungsi dengan nama sm.density.compare() yang dapat digunakan untuk menghasilkan fungsi kepekatan dari beberapa kelompok sekaligus, dan kemudian menampilkan kurva fungsi kepekatan dalam satu gambar.

Berikut ini adalah ilustrasi program untuk menghasilkan penduga fungsi kepekatan bagi peubah age (usia ibu saat melahirkan) pada data lowbwt (berupa dataframe dengan nama data.lahir) untuk tiga jenis race (suku bangsa) ibu.

```
library(sm)
sm.density.compare(data.lahir$age, data.lahir$race,
                   xlab="usia ibu (tahun)",
                   lwd=2)
colfill<-c(2:(2+length(levels(data.lahir$race))))
legend(locator(1), levels(data.lahir$race), fill=colfill)
```

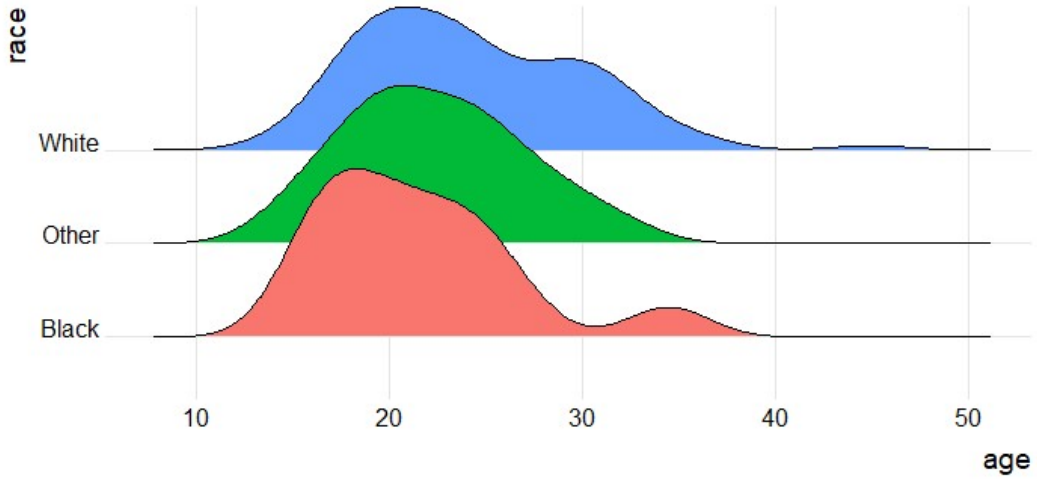
Program di atas akan menghasilkan grafik sebagai berikut. Ada beberapa informasi yang bisa kita peroleh. Pertama, terlihat bahwa puncak dari kurva berwarna merah

(sebaran usia ibu melahirkan dari suku bangsa Black) cenderung berada di sebelah kiri dari puncak dua kurva lainnya. Hal ini mengindikasikan bahwa secara rata-rata, ibu-ibu dari ras Black cenderung melahirkan pada usia lebih muda dibandingkan ibu dari ras lain. Informasi lain adalah tentang ukuran penyebaran. Tampak bahwa kurva biru (untuk ras White) cenderung lebih lebar dan lebih pendek daripada dua kurva lainnya. Ini menginformasikan bahwa keragaman usia ibu dari ras White cenderung lebih tinggi dibandingkan dua ras yang lainnya.

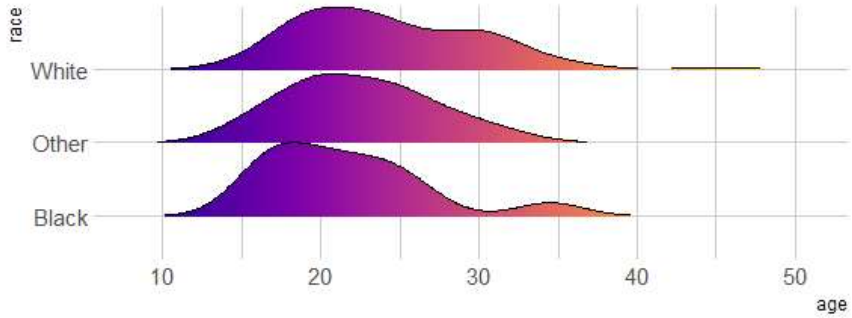


Berikut ini dua cara lain di R yang dapat digunakan untuk menghasilkan plot penduga fungsi kepekatan usia ibu dari tiga kelompok ras. Berbeda dengan sebelumnya, tiga plot fungsi kepekatan diletakkan pada sumbu yang disusun paralel, tidak dalam satu sumbu usia.

```
library(ggridges)
library(ggplot2)
ggplot(data.lahir, aes(x = age, y = race, fill = race)) +
  geom_density_ridges() +
  theme_ridges() +
  theme(legend.position = "none")
```




```
library(ggridges)
library(ggplot2)
library(viridis)
library(hrbrthemes)

# Plot
ggplot(data.lahir, aes(x = age, y = race, fill = ..x..)) +
  geom_density_ridges_gradient(scale=1, rel_min_height=0.01) +
  scale_fill_viridis(name = "usia", option = "C") +
  labs(title = '') +
  theme_ipsum() +
  theme(
    legend.position="none",
    panel.spacing = unit(0.1, "lines"),
    strip.text.x = element_text(size = 8)
  )
```

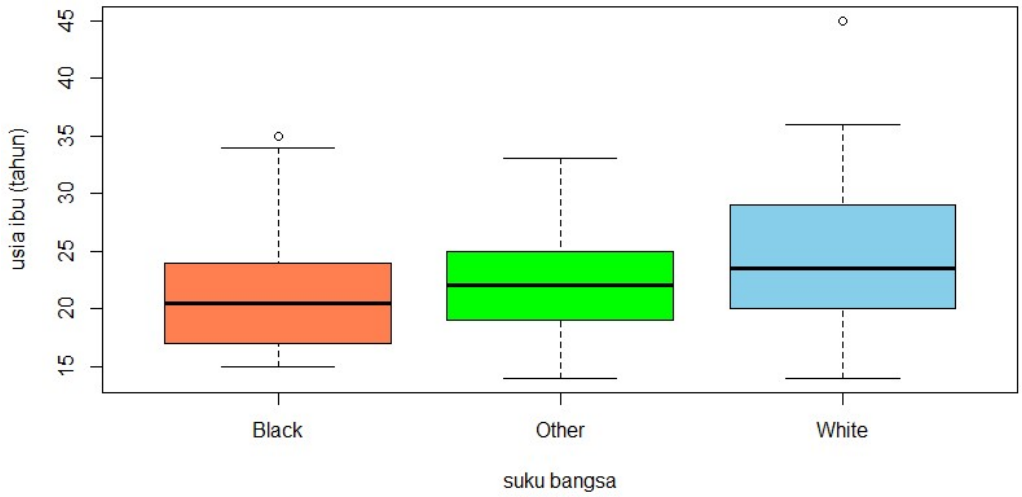


Visualisasi lain yang juga dapat digunakan untuk melakukan perbandingan sebaran dari beberapa (sub)populasi adalah dengan boxplot. Program R berikut ini dapat digunakan

untuk menghasilkan tiga buah boxplot masing-masing untuk tiga jenis ras ibu dari data yang sama dengan ilustrasi sebelumnya.

```
boxplot(data.lahir$age ~ data.lahir$race,
        col=c("coral", "green", "skyblue"),
        ylab="usia ibu (tahun)",
        xlab="suku bangsa")
```

Dari Gambar yang dihasilkan kita bisa melihat bahwa kotak dari boxplot ras Black berada pada posisi lebih rencah dibandingkan dua ras yang lainnya yang menunjukkan bahwa secara rata-rata usia ibu ras Black cenderung lebih muda. Informasi mengenai keragaman usia ibu pada ras White yang cenderung lebih besar yang diperoleh dari kurva fungsi kepekatan juga dapat terlihat disini dengan melihat bahwa kotak berwarna biru cenderung lebih besar ukurannya dibandingkan kotak ras White maupun Other.



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 $0 \leq k \leq 100$ 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% × 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 $0 \leq t \leq 1$ pada prinsipnya diperoleh sebagai data pada urutan ke t × n setelah datanya terlebih dahulu diurutkan.

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 $x_{(1)} \leq x_{(2)} \leq \cdots \leq x_{(n)}$.
- Setiap data yang terurut merupakan kuantil yang bersesuaian dengan fraksi

  $$p_i = \frac{i-1}{n-1} \text{ untuk i = 1, . . . , n}$$

- 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 adankan 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

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

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 (scatterplot) 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.

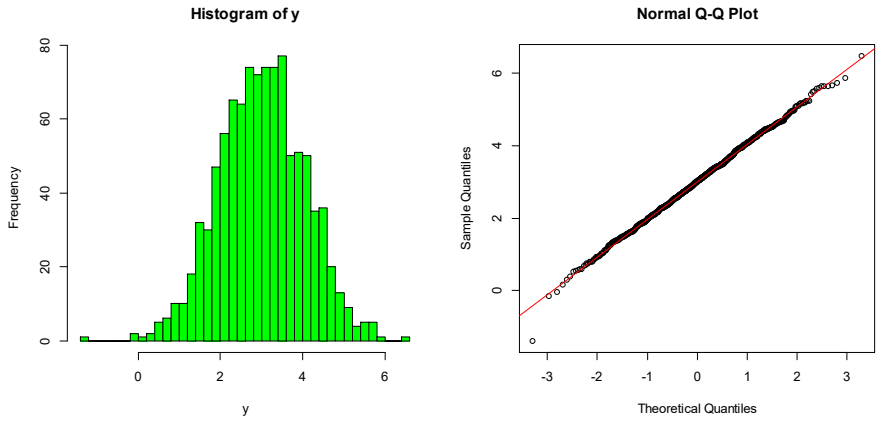
Tahapan pembuatan sebuah QQplot dari contoh berukuran n, {x1, x2, …, xn}, adalah sebagai berikut

- Urutkan data sehingga diperoleh susunan $x_{(1)} \leq x_{(2)} \leq \cdots \leq x_{(n)}$.
- Hitung pi = (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

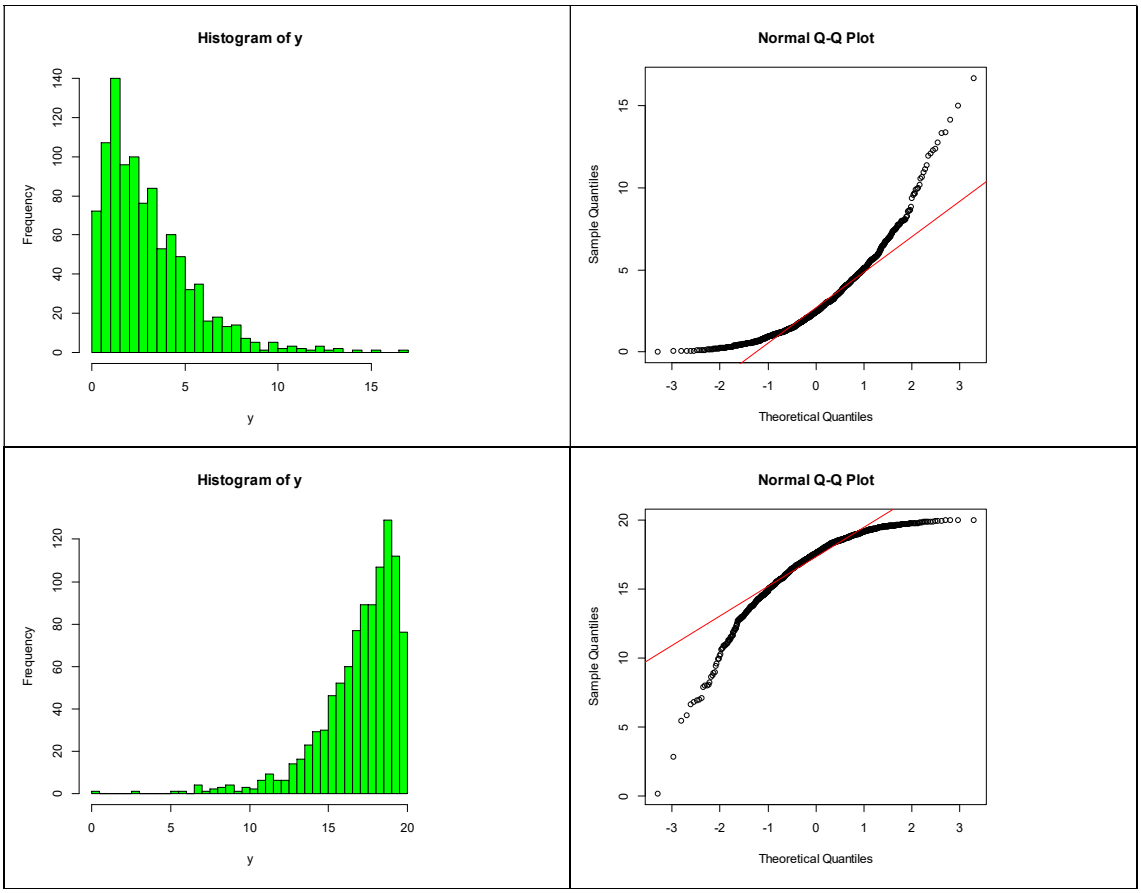
Berdasarkan prosedur umum yang disebutkan di atas, kita dapat membuat plot kuantil-kuantil normal (normal QQplot) dengan tahapan sebagai berikut:

- Urutkan data $x_{(1)} \leq x_{(2)} \leq \cdots \leq x_{(n)}$
- Hitung pi = (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. Berikut ini 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.

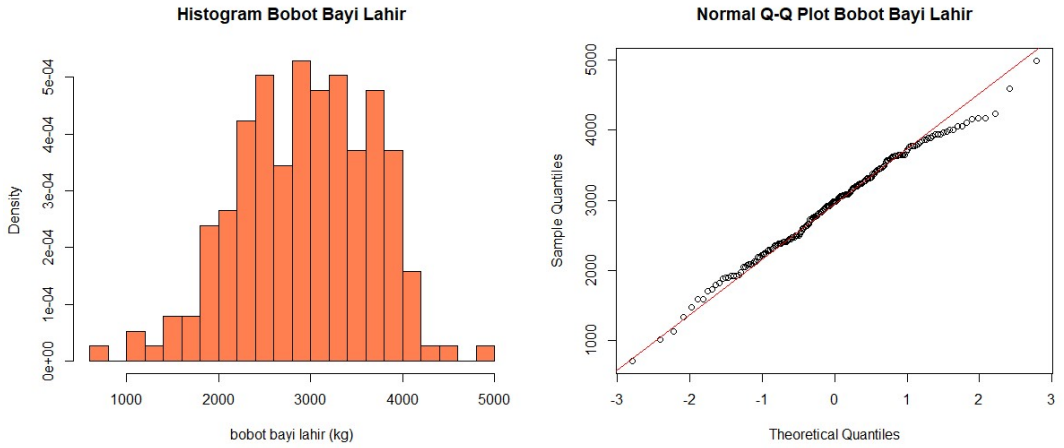


Jika kita memiliki data yang tidak berasal atau tidak mengikuti sebaran normal (misalnya menjulur ke kiri atau ke kanan) maka plot kuantil-kuantil normalnya tidak akan mengikuti garis lurus. Berikut ini adalah ilustrasi bentuk plot kuantil-kuantil normal dari data menjulur ke kanan dan ke kiri. Terlihat 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.



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.

```
data.lahir <- read.csv("D:/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")
```



Selain melakukan pemeriksaan secara visual menggunakan QQplot, terdapat juga uji fomral 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 diuuslkan 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 Oi. Menggunakan sebaran hipotetik yang akan dibandingkan, kita dapat menghitung peluang kejadian selang tersebut dan kemudian menghitung frekuensi harapannya (expected) yang dinotasikan Ei dan nilainya diperoleh dari Ei = pi × 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 yang pada ilustrasi ini dibagi menjadi 25 kelas. Selanjutna fungsi pnorm() digunakan untuk menghitung nilai peluang kumulatif normal yang nantinya akan diselisihkan menggukanan fungsi rollapply() sehingga diperoleh nilai peluang hipotetik. Perhatikan pada saat menggunakan fungsi pnorm() kita perlu mensupply nilai rata-rata dan simpangan baku sebaran yang dalam kasus ini diduga menggunakan rata-rata dan simpangan baku contoh.

```
data.lahir <- read.csv("D:/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 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 {x1, x2, …, xn} 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 x1, x2, …, xn adalah contoh acak, suatu fungsi sebaran kumulatif empris S(x) adalah sebuah fungsi dari x yang menyatakan proporsi banyaknya xi 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

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

Berikut ini adalah tabel nilai kritis dari dari uji Kolmogorov-Smirnov yang digunakan untuk menyimpulkan Tolak H0 jika statistik uji D melebihi nilai kritis dalam tabel.

Critical values for the Kolmogorov-Smirnov Test for goodness of fit

For completely specified continuous distributions:

n \ $1-\alpha$	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
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	$\frac{1.224}{\sqrt{n}}$	$\frac{1.358}{\sqrt{n}}$	$\frac{1.628}{\sqrt{n}}$

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).

```
data.lahir <- read.csv("D:/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
```