

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Sofitur Rohmah membuat model dengan melakukan peramalan menggunakan metode ARIMA pada jumlah penumpang kapal. Penelitian ini mengenai perkembangan transportasi laut Indonesia khususnya kapal penumpang, terjadi penurunan dalam 1 dekade terakhir dalam data penelitiannya, maka dari itu dilakukan sebuah peramalan menggunakan model ARIMA untuk mengetahui jumlah penumpang dalam perkembangan transportasi laut di Indonesia khususnya untuk kapal penumpang dengan perolehan jumlah secara kuantitatif.

Wahyuddin S dari lulusan UNIKOM juga melakukan penelitian menggunakan model ARIMA namun digabungkan dengan ANN untuk peramalan inflasi. Pada penelitian ini diusulkan sebuah metoda baru untuk prediksi inflasi memakai model ARIMA dan *Artificial Neural Network* (ANN). Data inflasi yang digunakan adalah data inflasi bulanan year-on-year dari tahun 2010 sampai dengan tahun 2018 yang diterbitkan oleh Badan Pusat Statistik (BPS).

Berikut ini beberapa Penelitian sebelumnya yang telah dilakukan :

Tabel 2. 1 Daftar Penelitian Terdahulu

| No. | Penelitian /Tahun | Judul | Persamaan | Perbedaan |
|-----|-------------------|--|---|--|
| 1. | 2018 | Implementasi Metode ARIMA dalam Peramalan Jangka Pendek (<i>Short Term Forecasting</i>) terhadap Jumlah Penumpang Kapal Eviews | Melakukan peramalan menggunakan metode ARIMA secara kuantitatif | <ol style="list-style-type: none"> 1. Objek penelitian berbeda 2. Hasil <i>output</i> yang dilakukan berbeda 3. Data variabel dalam melakukan peramalan berbeda |
| 2. | 2019 | Analisa Pengaruh Inflasi Terhadap Besar Sumbangan Pembinaan Pendidikan Menggunakan Algoritma Arima, Artificial Neural Network Dan Vector Autoregressive Model (Studi Kasus: Stmik Dipanegara Makassar) | Melakukan peramalan menggunakan metode ARIMA secara kuantitatif | <ol style="list-style-type: none"> 1. Objek penelitian berbeda 2. Hasil <i>output</i> yang dilakukan berbeda 3. Data variable dalam melakkan peramalan berbeda |

2.2 Definisi Peramalan

Peramalan (*forecasting*) yaitu prediksi nilai-nilai sebuah peubah berdasarkan kepada nilai yang diketahui dari peubah tersebut atau peubah yang berhubungan. Meramal juga dapat didasarkan pada keahlian keputusan (*judgement*), yang pada gilirannya didasarkan pada data historis dan pengalaman.

Perpaduan antara seni dan ilmu dalam memperkirakan keadaan di masa yang akan datang, dengan cara memproyeksikan data-data masa lampau ke masa yang akan datang dengan menggunakan model matematika maupun perkiraan yang subjektif disebut sebagai peramalan [2].

Peramalan adalah metode untuk memperkirakan suatu nilai di masa depan dengan menggunakan data masa lalu. Data inflasi merupakan salah satu data runtun waktu, dengan memodelkan data waktu lampau dapat digunakan untuk meramalkan data waktu mendatang. Dalam peramalan laju nilai inflasi dimana

data berupa runtun waktu, metode Seasonal Autoregressive Integrated Moving Average atau lebih dikenal dengan metode SARIMA dapat menunjukkan hasil peramalan yang mampu mengikuti pergerakan data aktual dari laju nilai inflasi. Sedangkan untuk metode Single Exponential Smoothing memiliki nilai akurasi yang baik pada data yang memiliki fluktuasi di sekitar rata-rata.

Peramalan adalah kegiatan untuk memperkirakan apa yang akan terjadi pada masa yang akan datang [3][4]. Peramalan merupakan bagian vital bagi setiap organisasi bisnis dan untuk setiap pengambilan keputusan manajemen yang sangat signifikan. Peramalan runtun waktu adalah serangkaian pengamatan terhadap suatu variabel yang diambil dari waktu ke waktu dan dicatat secara berurutan menurut urutan waktu [3][4].

2.2.1 Jenis-Jenis Metode Peramalan

Terdapat 2 pendekatan umum untuk jenis metode peramalan yaitu kualitatif dan kuantitatif. Metode peramalan kualitatif sangat penting dimana pada saat data historis tidak ada, tetapi metode ini bersifat sangat subjektif dan membutuhkan penilaian dari para ahli. Di lain pihak peramalan kuantitatif menggunakan data historis yang ada. Tujuan metode ini adalah mempelajari apa yang telah terjadi dimasa lalu untuk dapat meramalkan nilai-nilai dimasa yang akan datang [7].

a. Metode Peramalan Deret Berkala

Metode peramalan deret berkala, atau yang biasa disebut sebagai deret waktu (*time series*), merupakan salah satu metode yang termasuk dalam metode peramalan kuantitatif selain metode regresi atau kausal.

Metode peramalan deret berkala melibatkan proyeksi nilai yang akan datang dari sebuah variabel dengan berdasarkan seluruhnya pada pengamatan masa lalu dan sekarang dari variabel tersebut [7].

b. Metode Pemulusan

Metode pemulusan atau biasa disebut metode *smoothing*, termasuk dalam metode peramalan deret berkala. Metode pemulusan memiliki dasar metode yaitu pembobotan sederhana atau pemulusan pengamatan masa lalu dalam suatu sumber deret berkala untuk memperoleh ramalan masa mendatang [8].

Dalam pemulusan nilai-nilai historis ini, galat acak dirata-ratakan untuk menghasilkan ramalan “halus”. Diantara keuntungannya yaitu biaya yang rendah, mudah digunakan dalam penerapannya, dan cepat dalam penyampaiannya. Karakteristik ini dapat membuatnya menarik terutama bilamana horison waktunya relatif pendek (kurang dari 1 tahun).

Metode pemulusan terdiri atas metode pemulusan perataan, dimana pada saat melakukan pembobotan yang sama terhadap nilai-nilai pengamatan sesuai dengan pengertian konvensional tentang nilai tengah, dan metode pemulusan eksponensial menggunakan bobot berbeda untuk data masa lalu, karena bobotnya berciri menurun seperti eksponensial dari titik data yang terakhir sampai terawal.

2.2.2 Jenis-jenis Pola Data

Langkah penting dalam memilih suatu metode deret berkala (*time series*) yang tepat adalah dengan mempertimbangkan jenis pola data, sehingga metode yang paling tepat dengan pola tersebut dapat diuji.

Pola data dapat dibedakan menjadi empat jenis [5], yaitu:

1. Pola Horizontal atau *Horizontal Data Pattern*

Pola data ini terjadi bilamana data berfluktuasi di sekitar nilai rata-rata. Suatu produk yang penjualannya tidak meningkat atau menurun selama waktu tertentu termasuk jenis ini.

2. Pola Tren atau *Trend Data Pattern*

Pola data ini terjadi bilamana terdapat kenaikan atau penurunan suatu data dalam jangka panjang. Contohnya penjualan perusahaan, GNP (*Gross National Product*) dan berbagai indikator bisnis atau ekonomi lainnya, selama perubahan sepanjang waktu.

3. Pola Musiman atau *Seasonal Data Pattern*

Pola data ini terjadi bilamana suatu deret dipengaruhi oleh faktor musiman (misalnya kuartal tahun tertentu, bulan atau hari-hari pada minggu tertentu). Penjualan dari produk seperti minuman ringan, es krim dan bahan bakar pemanas ruang semuanya menunjukkan jenis pola ini.

4. Pola Siklis atau *Cycled Data Pattern*

Pola data ini terjadi bilamana datanya dipengaruhi oleh fluktuasi ekonomi jangka panjang seperti yang berhubungan dengan siklus bisnis. Contohnya penjualan produk seperti mobil, baja.

2.2.3 Jangka Waktu Peramalan

Jangka waktu peramalan dapat dikelompokkan menjadi tiga kategori [6], yaitu :

1. Peramalan jangka pendek, peramalan untuk jangka waktu kurang dari tiga bulan.
2. Peramalan jangka menengah, peramalan untuk jangka waktu antara tiga bulan sampai tiga tahun.
3. Peramalan jangka panjang, peramalan untuk jangka waktu lebih dari tiga tahun.

2.2.4 Proses Peramalan

Didalam melakukan proses peramalan, apapun bentuk dan jenis peramalan yang akan dilakukan, terdapat lima langkah proses peramalan yang bisa dilakukan [9], yaitu :

1. Formulasi masalah dan pengumpulan data.
Jika metode peramalan kuantitatif yang dipakai maka data yang relevan harus tersedia dan benar. Jika data yang sesuai tidak tersedia maka mungkin perumusan masalah perlu dikaji ulang atau memeriksa kembali metode peramalan kuantitatif yang dipakai.
2. Manipulasi dan pembersihan data
Ada kemungkinan kita memiliki terlalu banyak atau terlalu sedikit data yang dibutuhkan. Sebagian data mungkin tidak relevan pada masalah. Sebagian data mungkin memiliki nilai yang hilang yang harus diestimasi. Sebagian data mungkin harus dihitung dalam unit selain unit aslinya. Sebagian data mungkin harus diproses terlebih dahulu (misal, dijumlahkan dari berbagai sumber). Data yang lain kemungkinan sesuai tetapi hanya pada periode historis tertentu. Biasanya perlu usaha untuk mengambil data

dalam suatu bentuk yang di butuhkan untuk menggunakan prosedur peramalan tertentu.

3. Pembentukan dan evaluasi model

Pembentukan dan evaluasi model menyangkut kesesuaian data yang terkumpul pada suatu model peramalan yang sesuai dengan meminimalkan galat peramalan.

4. Implementasi model (peramalan sebenarnya)

Implementasi model terdiri dari model peramalan aktual yang dibuat ketika data yang sesuai telah terkumpul dan terpilihnya model peramalan yang sesuai. Peramalan untuk periode sekarang dengan nilai historis aktual diketahui sering kali digunakan untuk mengecek keakuratan dari proses.

5. Evaluasi peramalan

Implementasi model terdiri dari model peramalan aktual yang dibuat ketika data yang sesuai telah terkumpul dan terpilihnya model peramalan yang sesuai. Peramalan untuk periode sekarang dengan nilai historis aktual diketahui sering kali digunakan untuk mengecek keakuratan dari proses.

2.2.5 Karakteristik Peramalan

Karakteristik dari peramalan yang baik harus memenuhi beberapa kriteria yaitu dari hal-hal sebagai berikut:

1. Ketelitian/ Keakuratan

Tujuan utama peramalan adalah menghasilkan prediksi yang akurat. Peramalan yang terlalu rendah mengakibatkan kekurangan persediaan (inventory). Peramalan yang terlalu tinggi akan menyebabkan inventory yang berlebihan dan biaya operasi tambahan.

2. Biaya

Biaya untuk mengembangkan model peramalan dan melakukan peramalan akan menjadi signifikan jika jumlah produk dan data lainnya semakin besar. Mengusahakan melakukan peramalan jangan sampai menimbulkan ongkos yang terlalu besar ataupun terlalu kecil. Keakuratan peramalan dapat ditingkatkan dengan mengembangkan model lebih kompleks dengan konsekuensi biaya menjadi lebih mahal. Jadi ada nilai tukar antara biaya dan keakuratan.

3. Responsif

Ramalan harus stabil dan tidak terpengaruhi oleh fluktuasi demand.

4. Sederhana

Keuntungan utama menggunakan peramalan yang sederhana yaitu kemudahan untuk melakukan peramalan. Jika kesulitan terjadi pada metode sederhana, diagnosa dilakukan lebih mudah. Secara umum, lebih baik menggunakan metode paling sederhana yang sesuai dengan kebutuhan peramalan.

2.3 Autoregressive Integrated Moving Average (ARIMA)

2.3.1 Prinsip Dasar dan Tujuan Analisis

1. Prinsip Dasar

ARIMA sering juga disebut metode runtun waktu *Box-Jenkins*. ARIMA sangat baik ketepatannya untuk peramalan jangka pendek, sedangkan untuk peramalan jangka panjang ketepatan peramalannya kurang baik. Biasanya akan cenderung flat (mendatar/konstan) untuk periode yang cukup panjang [10].

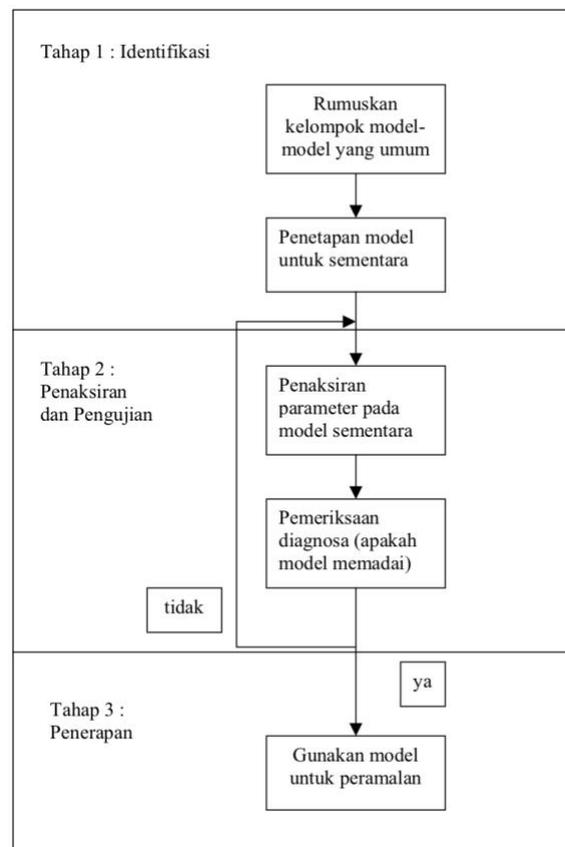
Model Autoregresif Integrated Moving Average (ARIMA) adalah model yang secara penuh mengabaikan variable independen dalam membuat peramalan. ARIMA menggunakan nilai masa lalu dan sekarang dari variabel dependen untuk menghasilkan peramalan jangka pendek yang akurat. ARIMA cocok jika observasi dari deret waktu (*time series*) secara statistik berhubungan satu sama lain (*dependent*).

2. Tujuan Analisis

Tujuan model ini adalah untuk menentukan hubungan statistik yang baik antar variabel yang diramal dengan nilai historis variabel tersebut sehingga peramalan dapat dilakukan dengan model tersebut.

2.3.2 Model Matematis

Model ARIMA terdiri dari tiga langkah dasar, yaitu tahap identifikasi, tahap penaksiran dan pengujian, dan pemeriksaan diagnostik. Selanjutnya model ARIMA dapat digunakan untuk melakukan peramalan jika model yang diperoleh memadai.



Gambar 2. 1 Skema Pendekatan Box Jenkins [10]

2.3.3 Kelebihan dan Kekurangan ARIMA

1. Kelebihan ARIMA

- Merupakan model tanpa teori karena variabel yang digunakan adalah nilai-nilai lampau dan kesalahan yang mengikutinya.

- b. Memiliki tingkat akurasi peramalan yang cukup tinggi karena setelah mengalami pengukuran kesalahan peramalan MAE (*mean absolute error*), nilainya mendekati nol.
- c. Cocok digunakan untuk meramal sejumlah variabel dengan cepat, sederhana, akurat dan murah karena hanya membutuhkan data variabel yang akan diramal [10].

2. Kekurangan ARIMA

- a. Untuk data peramalan dalam periode yang cukup panjang ketepatannya kurang baik karena biasanya akan cenderung flat (datar/konstan).
- b. ARIMA akan mengalami penurunan keakuratan apabila terdapat komponen nonlinier *time series* pada data pengamatan [10].

2.3.4 Stasioneritas dan Nonstasioneritas

Hal yang perlu diperhatikan adalah bahwa kebanyakan deret berkala bersifat nonstasioner dan bahwa aspek-aspek AR dan MA dari model ARIMA hanya berkenaan dengan deret berkala yang stasioner.

Stasioneritas berarti tidak terdapat pertumbuhan atau penurunan pada data. Data secara kasarnya harus horizontal sepanjang sumbu waktu. Dengan kata lain, fluktuasi data berada di sekitar suatu nilai rata-rata yang konstan, tidak tergantung pada waktu dan varians dari fluktuasi tersebut pada pokoknya tetap konstan setiap waktu.

Suatu deret waktu yang tidak stasioner harus diubah menjadi data stasioner dengan melakukan *differencing*. Yang dimaksud dengan *differencing* adalah menghitung perubahan atau selisih nilai observasi. Nilai selisih yang diperoleh dicek lagi apakah stasioner atau tidak. Jika belum stasioner maka dilakukan *differencing* lagi. Jika varians tidak stasioner, maka dilakukan transformasi logaritma.

Terdapat dua jenis stasioneritas, yaitu stasioner kuat atau stasioner orde pertama dan stasioner lemah atau stasioner orde kedua (Yaffee,2000).

a. Stasioner kuat (*strictly stationarity*)

Dikatakan memiliki stasioner yang kuat jika data runtun waktu memiliki rata-rata tetap dan varians tetap serta autokovarians yang konstan.

b. Stasioner lemah (*weakly stationarity*)

Untuk stasioner lemah, hanya dua karakteristik yang ditentukan, yaitu nilai rata-rata dan varians yang konstan. Untuk autokovarians stasioneritas jenis ini merupakan fungsi dari lag (nilai selisih waktu pengamatan).

Pada aplikasinya, khususnya untuk peramalan dengan menggunakan pemodelan ARIMA menggunakan stasioner lemah untuk tahapan prosesnya. Seperti yang telah disebutkan sebelumnya, dua syarat menjadi karakteristik stasioneritas jenis ini, berikut penjelasan lebih lengkapnya.

Data dikatakan stasioner $\{ Z_t \}$ jika memenuhi kedua syarat berikut ini (Box,2008).

a. Mean stationarity

Data bersifat stasioner terhadap rata-rata (*mean*) jika berfluktuasi di sekitar nilai rata-rata yang tetap dari waktu ke waktu seperti diformulasikan pada persamaan 2-1 berikut ini.

$$\mu = E[Z_t] = \int_{-\infty}^{\infty} Zp(Z)dZ \dots\dots\dots(1)$$

Persamaan 2- 1 Stasioneritas rata-rata

dengan,

Z_t : data ke-t

μ : rata-rata

$p(Z)$: distribusi probabilitas

b. *Variance stationarity*

Data bersifat stasioner terhadap varians jika berfluktuasi pada varian yang tetap dari waktu ke waktu yang diformulasikan pada persamaan 2-2 di bawah ini.

$$\sigma_z^2 = E[(Z_t - \mu)^2] = \int_{-\infty}^{\infty} Z - (Z - \mu)^2 p(Z) dZ \dots\dots\dots(2)$$

Persamaan 2- 2 Stasioneritas varians

dengan,

Z_t : data ke-t

σ_z^2 : varians data ke-t

μ : rata-rata

$p(Z)$: distribusi probabilitas

2.3.5 Stasioneritas dan *Differencing*

Hal yang perlu diperhatikan adalah bahwa kebanyakan deret berkala bersifat non-stasioner dan bahwa aspek-aspek AR dan MA dari model ARIMA hanya berkenaan dengan deret berkala yang stasioner. Stasioneritas berarti tidak terdapat pertumbuhan atau penurunan pada data. Data secara kasarnya harus horizontal sepanjang sumbu waktu. Dengan kata lain, fluktuasi data berada di sekitar suatu nilai rata-rata yang konstan, tidak tergantung pada waktu dan varians dari fluktuasi tersebut pada pokoknya tetap konstan setiap waktu.

Menurut Makridakis,dkk (1999: 351) Stasioneritas mempunyai makna bahwa tidak terdapat pertumbuhan atau penurunan pada data. Dengan kata lain, fluktuasi data berada di sekitar suatu nilai rata-rata yang konstan, tidak tergantung pada waktu dan varians dari fluktuasi tersebut. Plot data deret berkala dapat digunakan untuk mengetahui suatu data telah stasioner atau belum. Demikian juga plot autokorelasi juga dapat memperlihatkan kestasioneritasan suatu data deret berkala.

Suatu deret waktu yang tidak stasioner harus diubah menjadi data stasioner dengan melakukan *differencing*. Yang dimaksud dengan *differencing* adalah menghitung perubahan atau selisih nilai observasi. Nilai selisih yang diperoleh dicek lagi apakah stasioner atau tidak. Jika belum Stasioner maka dilakukan *differencing* lagi. Jika varians tidak stasioner, maka dilakukan transformasi logaritma.

Jika proses pembangkitan yang mendasari suatu deret berkala didasarkan pada nilai tengah konstan dan varians konstan, maka deret berkala berupa stasioner, dimana artinya bahwa sebuah deret sudah stasioner jika sifat

stastistiknya bebas dari periode selama pengamatan. Jadi, Stasioneritas adalah tidak terdapat pertumbuhan atau penurunan pada data dan harus horisontal sepanjang waktu. Dengan kata lain, fluktuasi data berada di sekitar suatu nilai rata-rata yang konstan, tidak tergantung pada waktu dan varians dari fluktuasi tersebut serta tetap konstan setiap waktu.

Dalam metode deret berkala (*time series*) pengujian kestasioneran data sangat diperlukan, dimana apabila data tersebut sudah stasioner maka dapat digunakan untuk melakukan peramalan di masa yang akan datang. Ada beberapa hal yang diperlukan untuk melihat suatu data telah stasioner antara lain sebagai berikut :

- a. Apabila suatu deret berkala diplot, dan kemudian tidak terbukti adanya perubahan nilai tengah dari waktu ke waktu, maka dikatakan bahwa deret tersebut stasioner pada nilai tengahnya.
- b. Apabila plot deret berkala tidak memperlihatkan adanya perubahan yang jelas dari waktu ke waktu, maka dapat dikatakan bahwa deret berkala tersebut adalah stasioner pada variasinya.
- c. Apabila plot deret berkala memperlihatkan adanya penyimpangan nilai tengah atau terjadi perubahan varian yang jelas dari waktu ke waktu, maka dikatakan bahwa deret berkala tersebut mempunyai nilai tengah yang tidak Stasioner atau mempunyai nilai variasi yang tidak stasioner.
- d. Apabila plot deret berkala memperlihatkan adanya penyimpangan pada nilai tengah serta terjadi perubahan varians dari waktu ke waktu, maka dikatakan bahwa deret data tersebut mempunyai nilai tengah dan variasi yang tidak stasioner

Untuk melakukan peramalan dengan menggunakan metode deret berkala Box-Jenkins, maka dipilih deret berkala yang stasioner baik nilai tengahnya maupun variasinya, sehingga untuk deret berkala yang tidak stasioner baik nilai tengah maupun variasinya perlu dilakukan suatu proses untuk mendapatkan keadaan yang stasioner. Proses untuk mendapatkan keadaan stasioner nilai tengah adalah dengan melakukan *difference* (pembedaan), sedangkan untuk mendapatkan keadaan stasioner varians perlu dilakukan transformasi. Kedua hal tersebut bisa

dilakukan salah satu saja atau kedua-duanya, tergantung dari keadaan stasioneritas dari deret data deret berkala yang akan dipilih untuk peramalan.

Pembedaan adalah usaha untuk menstabilkan nilai tengah dari deret berkala. Proses pembedaan bisa dilakukan beberapa kali yang biasanya disebut dengan pembedaan order ke-d, sehingga bila melakukan pembedaan satu kali maka disebut *difference* order-1, bila dilakukan pembedaan dua kali maka *difference* order-2 dan seterusnya. Namun pembedaan yang biasa dilakukan paling tinggi adalah sampai dengan orde-2 saja, karena bila dilakukan pembedaan lebih dari order-2 maka deret berkala akan semakin mendekati linier, sehingga sifat-sifat deret berkala akan hilang.

Yang dimaksud dengan *differencing* adalah menghitung perubahan atau selisih nilai observasi. Persamaan *differencing* sebagai berikut (Hanke, 2009):

$$X_t' = X_t - BX_t \dots\dots\dots(3) (2.6)$$

Keterangan :

X_t' : nilai deret berkala setelah *differencing*

X_t : nilai deret berkala pada waktu t

BX_t : orde *differencing*

Suatu deret berskala dikatakan stasioner atau menunjukkan kesalahan acak adalah jika koefisien autokorelasi untuk semua lag, yaitu angka yang ditunjukkan pada setiap interval secara statistik tidak berbeda dari nol atau berbeda dari nol hanya untuk beberapa lag yang di depan. Suatu koefisien autokorelasi dikatakan tidak berbeda dari nol.

2.3.6 Autocorrelation Function (ACF) dan Fungsi Autokorelasi Parsial (PACF)

Model peramalan Box-Jenkins diidentifikasi dengan mengamati perilaku dari *Autocorrelation Function* (ACF) dan *Partial Autocorrelation Function* (PACF). Pada dasarnya baik ACF maupun PACF memiliki fungsi yang sama yaitu untuk menunjukkan hubungan linear antar data-data *time series* yang dipisahkan oleh lag-k unit waktu sama yaitu untuk menunjukkan hubungan linear antar data-data *time series* yang dipisahkan oleh lag-k unit waktu.

Tabel 2. 2 *Rangkuman perintah log dan log return*

| Perintah | Definisi | Persamaan Matematis |
|-------------|---|---|
| log(X) | Untuk melakukan transformasi logaritma terhadap X_t | $\log(X)$ |
| dlog(X,n) | Untuk melakukan diferensi order n terhadap data $\log(X_t)$ | $\Delta^n \log(X_t) = \Delta^n (\log(X_t) - \log(X_{t-1}))$ |
| dlog(X,n,s) | Untuk melakukan diferensi order n dan diferensi musiman terhadap data $\log(X_t)$ | $\Delta^n \Delta_s \log(X_t) = (1 - B)^n B^s \log(X_t)$ |

a. Fungsi Autokorelasi

Autokorelasi adalah hubungan yang terjadi antara anggota-anggota data runtun waktu melalui pengamatan yang dilakukan (Makridakis, 1999). Jika tidak terdapat autokorelasi dalam data, maka dapat dikatakan bahwa data tersebut random atau tidak memiliki pola.

Nilai ACF pada lag-k dapat diformulasikan sebagai berikut.

$$\rho_k = \frac{\sum_{t=1}^{N-k} (Z_t - \bar{Z})(Z_{t+k} - \bar{Z})}{\sum_{t=1}^N (Z_t - \bar{Z})^2} \dots\dots\dots(4)$$

Persamaan 2- 7 Fungsi autokorelasi

dengan,

ρ_k : fungsi autokorelasi

Z_t : data waktu-t

\bar{Z} : rata-rata data ke-t

Z_{t+k} : data waktu t+k

Berikut contoh dari suatu fungsi autokorelasi dari suatu data runtun waktu dapat dilihat pada Gambar 2. 2

b. Fungsi Parsial Autokorelasi

Fungsi autokorelasi parsial menunjukkan tingkat keeratan antara Z_t dan Z_{t+k} dengan syarat menghilangkan pengaruh dari lag 1,2, dan seterusnya sampai k-1. Nilai PACF pada lag-k dapat ditentukan sebagai berikut.

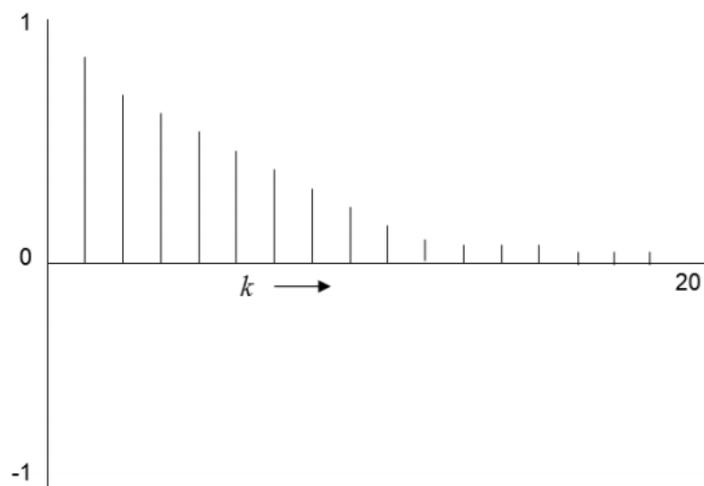
$$\phi_{kk} = \frac{\rho_k - \sum_{j=1}^{k-1} \phi_{k-1,j} \rho_{k-j}}{1 - \sum_{j=1}^{k-1} \phi_{k-1,j} \rho_{k-j}} \dots\dots\dots (5)$$

Persamaan 2- 8 Fungsi Parsial Autokorelasi

Dengan,

ϕ_{kk} : fungsi autokorelasi parsial

ρ_k : fungsi autokorelasi



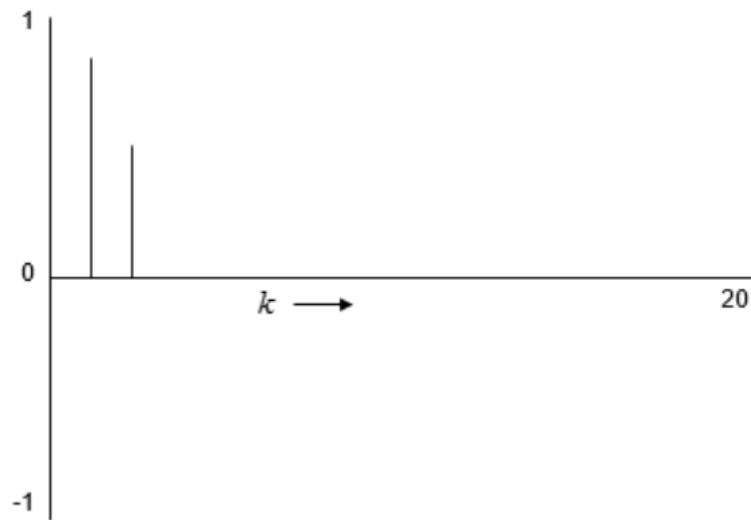
Gambar 2. 2 Fungsi autokorelasi suatu data Z_t

Contoh salah satu bentuk PACF teoritik dari suatu data runtun waktu dapat dilihat pada Gambar 2. 3 sebagai berikut.

Melalui perhitungan ACF dan PACF dapat ditentukan model *autoregressive* (AR) atau model *moving average* (MA) orde ke berapa data yang sedang dianalisis. Tabel 2.3 berikut adalah tabel yang secara umum dapat digunakan untuk mengidentifikasi suatu data *time series* berdasarkan bentuk ACF dan PACF taksirannya.

2.3.7 Model-Model Stasioner

Model-model Stasioner dibagi ke dalam 3 kelompok, yaitu metode *Autoregressive (AR)*, *Moving Average (MA)*, dan model campuran *Autoregressive and Moving Average* yang mempunyai karakteristik dari dua model pertama.



Gambar 2. 3 Fungsi parsial autokorelasi suatu data Z_t

Tabel 2. 3 Pola teoritik ACF dan PACF dari proses stasioner

| Proses ACF | Sampel ACF | Sampel PACF |
|--------------------|--|--|
| <i>White noise</i> | Tidak ada yang melewati batas interval pada lag >0 | Tidak ada yang melewati batas interval pada lag >0 |
| AR(p) | Meluruh menuju 0 secara eskponensial | Diatas batas interval maksimum sampai pada lag ke p dan di bawah batas pada lag $>p$ |
| MA(q) | Diatas batas interval maksimum sampai pada lag ke q dan di bawah batas pada lag $>q$ | Meluruh menuju 0 secara eskponensial |
| ARMA(p,q) | Meluruh menuju 0 secara eskponensial | Meluruh menuju 0 secara eskponensial |

a. *Autoregressive (AR)*

Proses autoregresif sesuai dengan namanya adalah proses hasil regresi dengan dirinya sendiri. Bentuk umum dari proses autoregresif tingkat p atau AR(p) adalah sebagai berikut.

$$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + a_t \dots\dots\dots(6)$$

Persamaan 2- 9 Bentuk umum AR (p)

dimana,

Z_t = data ke-t

ϕ_p = parameter autoregressive ke-t

a_t = nilai error pada saat ke-t

b. *Moving Average (MA)*

Bentuk umum dari proses *moving average* tingkat q atau MA(q) didefinisikan sebagai

$$Z_t = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_p a_{t-p} \dots\dots\dots(7)$$

Persamaan 2- 10 Bentuk umum MA (q)

dimana,

Z_t = data ke-t

θ_p = parameter moving average ke-t

a_t = nilai error pada saat ke-t

c. *Autoregressive and Moving Average (ARMA)*

Pada Metode ARMA orde p dan q (AR(p) dan MA(q)) adalah gabungan antara Autoregressive Model (AR) dan Moving Average (MA). Berikut ini merupakan rumus dari ARMA.

$$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q} \dots\dots\dots(8)$$

Persamaan 2- 11 Bentuk umum ARMA

dimana,

Z_t = data ke-t

ϕ_p = parameter autoregressive ke-t

a_t = nilai error pada saat ke-t

θ_p = parameter moving average ke-t

2.3.8 Model-Model Tak Stasioner

Model-model tak stasioner dapat ditandai dengan adanya proses *differencing* pada pemodelannya. Proses *differencing* (d) mengindikasikan bahwa data telah dilakukan transformasi hingga mencapai stasioner. Model tak stasioner dapat dijelaskan pada paparan sebagai berikut.

a. Model ARIMA (p,d,q)

Suatu time series yang dihasilkan dari 3 proses meliputi autoregressive (p), moving average (q), dan integrated (d) yang menandakan telah dilakukan proses differencing. Pemodelan ini dituliskan sebagai berikut

$$Z_t = (1 + \phi_1)Z_{t-1} + (\phi_2 - \phi_1)Z_{t-2} + \dots + (\phi_p - \phi_{p-1})Z_{t-p} - \theta_p Z_{t-p-1} + a_t - \theta_1 a_{t-1} - \dots - \theta_q a_{t-q} \dots \dots \dots (9)$$

Persamaan 2- 12 Bentuk umum ARIMA

dimana,

Z_t = data ke-t

ϕ_p = parameter autoregressive ke-t

a_t = nilai error pada saat ke-t

θ_p = parameter moving average ke-t

b. Model SARIMA (Seasonal ARIMA)

Selain ditandai dengan adanya proses diferensi, model tak stasioner ditandai dengan adanya faktor musiman. Musiman didefinisikan sebagai suatu pola yang berulang sendiri setelah interval waktu yang tetap. Pola musiman dapat berupa triwuanan (3 bulanan), kwartalan (4 bulanan), semesteran (6 bulanan), atau tahunan (12 bulanan). Contoh sederhananya penjualan pakaian akan melonjak ketika menjelang Hari Raya Keagamaan. Jika pola tersebut konsisten, koefisien autokorelasi dari lag 12 bulan akan mempunyai nilai positif yang besar, menunjukkan adanya faktor musiman.

Untuk data stasioner, faktor musiman dapat ditentukan dengan mengidentifikasi koefisien autokorelasi lebih dari dua atau tiga timelag yang signifikan berbeda dari nol. Faktor musiman tidak selalu mudah diidentifikasi jangka musiman tersebut berkombinasi dengan pola lain seperti trend. Semakin kuatnya pengaruh trend (data tak stasioner), maka

semakin tidak jelaslah faktor musimannya. Dengan demikian, data harus ditransformasi ke bentuk stasioner sebelum menentukan faktor musimannya.

Untuk membuat model bagi data musiman, secara umum dapat digunakan dua jenis model sebagai berikut.

1. Model Seasonal Additive

Pada model ini, komponen musiman berinteraksi dengan komponen non-musiman dalam model secara additive, dinyatakan sebagai model SARIMA ((p,P,(d,D),(q,Q))s (Seasonal ARIMA) bagi proses X_t , yang dapat ditulis sebagai berikut.

$$(1 - a_1B - \dots - a_pB^p)(1 - B)^d(1 - B^s)^D X_t = (1 + b_1B + \dots + b_qB^q + \theta_1B^s + \dots + \theta_QB^{sQ})\varepsilon_t \dots(10)$$

Persamaan 2- 13 Model SARIMA additive moving average

dengan

B = operator backward

p dan q = orde komponen non-musiman dari model ARIMA

d = orde diferensi komponen non-musiman

P = orde koefisien autoregresi dari komponen musiman multiplikatif

D = orde diferensi musiman

Q = orde koefisien moving verage dari komponen musiman multiplikatif

Untuk model diatas, komponen musiman digambarkan dengan menggunakan komponen moving average, komponen musiman dapat juga dinyatakan dengan menggunakan komponen autoregresif, yakni menggunakan model sebagai berikut.

$$(1 - a_1B - \dots - a_pB^p - \beta_1B^s - \dots - \beta_QB^{sQ})(1 - B)^d(1 - B^s)^D X_t = (1 + b_1B + \dots + b_qB^q)\varepsilon_t \dots(11)$$

Persamaan 2- 14 Model SARIMA additive autoregressive

2. Model Seasonal multiplikatif

Pada model ini komponen berinteraksi dengan komponen nonmusiman dalam model secara multiplikatif berbentuk perkalian, dapat ditulis sebagai berikut.

$$(1 - a_1B - \dots - a_pB^p)(1 - B)^d(1 - Bs)Dxt \\ = (1 + b_1B + \dots + b_qB^q)(1 + \theta_sB^s + \dots + \theta_QB^sQ)\epsilon_t \dots (12)$$

Persamaan 2- 15 Model SARIMA multiplikatif moving average

Disini komponen musiman digambarkan menggunakan komponen moving average orde s. komponen musiman dapat juga dinyatakan dengan menggunakan komponen autoregressive, yakni sebagai berikut.

$$(1 - a_1B - \dots - a_pB^p)(1 - B)^d(1 - Bs)D(1 - \beta_sB^s + \dots - \beta_pB^sP)xt \\ = (1 + b_1B + \dots + b_qB^q)\epsilon_t \dots (13)$$

Persamaan 2- 16 Model SARIMA multiplikatif autoregressive

Model multiplikatif ini dapat dinotasikan sebagai model ARIMA (p,d,q)(P,D,Q)s.

2.3.9 Diagnostic Checking

Diagnostic check dari model yang telah diestimasi dilakukan dengan memverifikasi kesesuaian model dengan sifat-sifat data. Jika model dugaan yang didapatkan akan memiliki sifat-sifat yang mirip dengan data asli. Dengan demikian residual yang dihitung berdasarkan model yang telah diestimasi mengikuti asumsi error dari model yang teoretis, seperti sifat *white noise*. Untuk melihat apakah residual bersifat white noise dapat dilakukan dengan dua cara, yaitu sebagai berikut.

- Melihat apakah plot sampel ACF/PACF residual yang terstandarisasi (residual dibagi estimasi standar deviasi residual) telah memenuhi sifat-sifat proses white noise dengan mean 0 dan variansi 1.
- Melakukan uji korelasi serial, yakni menguji hipotesis : $H_0: \rho_1 = \rho_2 = \dots = \rho_k, k < n$ tidak terdapat korelasi serial dalam residual lag-k, k (p + q). Disini menunjukkan nilai sampel ACF pada lag-j sedangkan p dan q menunjukkan order dari model ARMA (p,q). Apabila hipotesis *diagnostic*

check ditolak maka model dugaan yang telah didapatkan tidak dapat digunakan, dan selanjutnya dapat diidentifikasi kembali model yang sesuai dengan data. Selanjutnya dalam praktik mungkin banyak model yang memenuhi pengujian *diagnostic* diatas. Untuk memilih model terbaik diantara model-model yang memenuhi uji diagnostik, dapat dipilih model yang meminimumkan ukuran kriteria informasi seperti Akaike Information Criteria, $AIC = n \ln(\sigma^2) + 2(p + q + 1)$, $\sigma^2 = SSE/n$, dengan SSE = Sum of Squared Error yang dapat diestimasi dari jumlahan kuadrat semua nilai residual. Akan tetapi, diketahui untuk model autoregressive, kriteria AIC tidak memberikan order p yang konsisten, sehingga untuk pembandingan dapat digunakan kriteria informasi lain, seperti Schwartz Bayesian Information Criteria, $SBC = n \ln(\sigma^2) + 2(p + q + 1)$, $n \ln$, ataupun bentuk kriteria informasi lain yang diusulkan.

2.3.10 Forecast dalam EViews

Dalam EViews terdapat dua jenis peramalan yaitu sebagai berikut.

a. *Static Forecast*

Dengan metode *static forecast* dilakukan peramalan satu langkah ke depan dari variabel dependen. Untuk setiap nilai peramalan satu langkah, EViews akan menghitung nilai peramalan dengan menggunakan nilai-nilai aktual dari nilai lag untuk variabel dependen yang digunakan untuk peramalan.

b. *Dynamic Forecast*

Dengan menggunakan metode *dynamic forecasting*, EViews akan melakukan peramalan n -langkah dari variabel dependen, dimulai dari waktu awal, yang dispesifikasikan pada range sampel peramalan. Untuk menjelaskan perbedaan antara metode dinamik dan static, akan digunakan contoh sederhana berikut ini.

Andaikan dimiliki model $Y_t = aY_{t-1} + X_t$ dengan Y menunjukkan variabel dependen dalam model, sedangkan X adalah variabel independen. Andaikan dimiliki data $X_1, X_2, \dots, X_n, X_{n-1}, \dots, X_{n+k}$ dan Y_1, Y_2, \dots, Y_n dan ingin melakukan peramalan data $Y_{n+1}, Y_{n+2}, \dots, Y_{n+k}$, $k \geq 2$ dan

dispesifikasikan rentang sampel untuk peramalan sebagai $s, \dots, n+k$, dengan $s \geq n+1$. Dengan menggunakan metode statik, maka untuk setiap waktu $s, \dots, n+k$ akan dilakukan peramalan atau perhitungan nilai Y satu langkah ke depan, sampai waktu ke $n+1$ yang akan dihitung sebagai $Y^{k+1} = aY^k + X^{k+1}$, $k = s, \dots, n$. Sedangkan untuk metode dinamik, untuk perhitungan nilai Y^{s+1} akan digunakan nilai sesungguhnya dari nilai Y^s , yakni dihitung menggunakan persamaan $Y^{s+1} = aY^s + X^{s+1}$. Sedangkan untuk periode selanjutnya perhitungan nilai Y^k , $k = s+2, \dots, n+k$ akan menggunakan nilai peramalan untuk Y yang diperoleh dari langkah sebelumnya, menggunakan rumus $Y^{k+1} = aY^k + X^{k+1}$, $k = s+1, \dots, n+k-1$.

Dengan memperhatikan penjelasan diatas, dapat disimpulkan bahwa metode *static forecasting* merupakan metode yang baik untuk melakukan *fitting* model (menghitung variabel dependen menggunakan estimator dari parameter data-data yang bersesuaian) dan *forecasting* satu langkah kedepan (dari data terakhir yang dimiliki) dengan cara menspesifikasikan rentang indeks waktu dari *forecasting* sebagai $1, \dots, n+1$. Sedangkan metode *dynamic forecasting* adalah baik untuk melakukan *forecasting* k langkah kedepan, $k \geq 1$, dengan cara menspesifikasikan rentang indeks waktu dari *forecasting* sebagai $n+1, \dots, n+k$.

2.3.11 Hubungan Akurasi Forecast

Menurut Jay Heinzer dan Barry Render (2009) ada beberapa perhitungan yang biasa dipergunakan untuk menghitung kesalahan peramalan (*forecast error*) total. Perhitungan ini dapat dipergunakan untuk membandingkan model peramalan yang berbeda, juga untuk mengawasi peramalan, untuk memastikan peramalan berjalan dengan baik. Model-model peramalan yang dilakukan kemudian divalidasi menggunakan sejumlah indikator. Indikator-indikator yang umum digunakan adalah rata-rata penyimpangan absolut (*Mean Absolute Deviation*), dan rata-rata kuadrat terkecil (*Mean Square Error*). [11]

Mean Squared Error (MSE) adalah metode lain untuk mengevaluasi metode peramalan. Masing-masing kesalahan atau sisa dikuadratkan. Kemudian dijumlahkan dan dibagi dengan jumlah observasi. Pendekatan ini mengatur kesalahan peramalan yang besar karena kesalahan-kesalahan itu dikuadratkan.

Metode itu menghasilkan kesalahan-kesalahan sedang yang kemungkinan lebih baik untuk kesalahan kecil, tetapi kadang menghasilkan perbedaan yang besar. *Mean squared error* adalah rata-rata dari kesalahan *forecast* dikuadratkan, atau jika dituliskan dalam bentuk rumus adalah: [11]

$$MSE = \frac{\sum_{t=1}^n (d_t - d'_t)^2}{n} \dots\dots\dots(14)$$

2.4 Penjualan

Suatu Perusahaan baik barang ataupun jasa, strategi penjualan sangat diperlukan untuk mendapatkan keuntungan yang besar. Salah satu cara yang dilakukan untuk strategi penjualan ini adalah dengan melakukan prediksi penjualan. Peramalan penjualan (*sales forecasting*) ialah teknik proyeksi permintaan langganan yang potensial untuk suatu waktu tertentu dengan berbagai asumsi [11].

2.5 Software R

R adalah suatu kesatuan *software* yang terintegrasi dengan beberapa fasilitas untuk manipulasi, perhitungan dan penampilan grafik yang handal. R berbasis pada bahasa pemrograman S, yang dikembangkan oleh AT&T Bell Laboratories sekarang Lucent Technologies pada akhir tahun '70 an. R merupakan versi gratis dari bahasa R dari *software* (berbayar) yang sejenis yakni S-PLUS yang banyak digunakan para peneliti dan akademisi dalam melakukan kegiatan ilmiahnya.

Pada awalnya, versi pertama R dibuat oleh Ross Ihaka and Robert Gentleman dari Universitas Auckland, namun selanjutnya R dikembangkan oleh tim yang disebut tim inti. Tim inti (*core team*) terdiri dari ahli statistik, ahli komputer & pemrograman, geografi, ekonomi dari institusi yang berbeda dari seluruh dunia yang mencoba membangun sebuah sistem (*software*) yang handal namun dengan biaya yang sangat murah. R dapat secara gratis di *download* dan digunakan dengan berlisensi pada GNU (*General Public License*).

Menurut kutipan dari penghargaan *Association for Computing Machinery Software* bagi John Chamber 1998, menyatakan bahwa bahasa pemrograman dapat “memanipulasi, visualisasi dan menganalisis data”. R dibuat searah dengan ide yang ada pada bahasa pemrograman S.

2.5.1 R dan Program Statistik Lainnya

Seperti dijelaskan sebelumnya, R merupakan “kerabat” dekat dari S-PLUS dimana secara fungsi dan sintaks/tata bahasa sama-sama menggunakan bahasa R, namun tidak identik. R dapat berinteraksi dengan program statistik, manipulasi, perhitungan dan penampilan grafik lainnya.

R adalah bahasa fungsional dimana terdapat inti bahasa yang menggunakan bentuk standar notasi aljabar, yang memungkinkan perhitungan numerik seperti $2+3$, atau 3^{11} . Selain itu tersedia pula fasilitas perhitungan dengan menggunakan fungsi.

Dengan beberapa fitur tersebut, R menjadi alat yang tangguh bagi para statistikawan, ahli ekonomi, peneliti dalam membantu risetnya, dikarenakan R dibangun dan didukung dengan model dan teori statistik terdepan dan menggunakan standar tertinggi bagi analisis data. R hampir dapat digunakan untuk berbagai bidang, mulai dari kalkulasi biasa seperti kalkulator, statistik, ekonometri, geografi, hingga pemrograman komputer.

2.5.2 Kelebihan dan Fitur-fitur R

R mempunyai karakteristik tersendiri, dimana selalu dimulai dengan prompt “>” pada console-nya. R mempunyai beberapa kelebihan dan fitur-fitur yang canggih dan berguna, diantaranya:

- a. Efektif dalam pengelolaan data dan fasilitas penyimpanan. Ukuran file yang disimpan jauh lebih kecil dibanding *software* lainnya.
- b. Lengkap dalam operator perhitungan *array*.
- c. Lengkap dan terdiri dari koleksi tools statistik yang terintegrasi untuk analisis data, diantaranya, mulai statistik deskriptif, fungsi probabilitas, berbagai macam uji statistik, hingga *time series*.
- d. Tampilan grafik yang menarik dan fleksibel ataupun *costumized*.

Dapat dikembangkan sesuai keperluan dan kebutuhan dan sifatnya yang terbuka, setiap orang dapat menambahkan fitur-fitur tambahan dalam bentuk paket ke dalam *software* R.

Selain kelebihan dan kelengkapan fitur-fiturnya, hal yang terpenting lainnya yakni, R bersifat dapat diinstall dan digunakan baik pada system operasi *Windows* , UNIX/LINUX maupun pada *Macintosh*. Untuk dua sistem operasi disebutkan terakhir diperlukan sedikit penyesuaian. Selain kelebihan disebutkan di atas, R didukung oleh komunitas yang secara aktif saling berinteraksi satu sama lain melalui Internet dan didukung oleh manual atau *R- help* yang menyatu pada *software* R. Sebagai catatan, tesis ini mengambil contoh pada penggunaan R pada system berbasis *Windows*.

2.5.3 R, Riset dan Akademis

Software R sangat cocok untuk riset, baik statistik, ekonomi, komputasi numerik dan pemrograman komputer. Karena didukung oleh banyak tenaga ahli dibidangnya, R layak dijadikan suatu perangkat lunak acuan bagi berbagai kalangan, terlebih di kalangan akademik (dosen, mahasiswa). Selain itu R memiliki fitur yang lengkap dan handal serta faktor tanggung jawab moral dan legal/hukum bukan lagi menjadi kekhawatiran dalam penggunaannya, karena dapat diperoleh secara gratis [20].

Berikut adalah beberapa contoh yang didapat dari R sebagai acuan implementasi pada:

1. Pemodelan matematis seperti *software* MATLAB dalam membentuk perspektif, cocok untuk jurusan teknik arsitek, sipil, mesin, dan ilmu komputer (pencitraan).
2. Pencitraan dan analisis kontur, cocok untuk jurusan geografi dan sejenis.
3. Proses analisis data statistik, dengan tampilan grafik plot yang *costumized* dan grafik fungsi densitas yang dapat diparalelkan dengan histogram. Cocok untuk bidang statistika, ekonomi, dan lain lain.