



## Pemanfaatan AI Perwakilan Manajer dalam Implementasi Persetujuan Elektronik Berbasis SVM di Bidang SIM-SDM

Arman Yusuf

Magister Manajemen, Fakultas Ekonomi dan Bisnis, Universitas Budi Luhur

<sup>1</sup>[arman.yusuf@budiluhur.ac.id](mailto:arman.yusuf@budiluhur.ac.id)

### Abstract

Artificial Intelligent (AI) has been popular and applied in many areas nowadays. Many features can be built upon machines by learning the characters human did, and later to inference the tasks based on what it has been learned before. Human tasks that have been learned by machine can then be utilized to replace human itself in repetitive tasks. In this research journal, the researcher utilizes one of the AI technology, Support Vector Machine (SVM) based Machine Learning to learn manager's characters in the approval activities. This later will be used by machine that acts as a manager in making approvals. To learn the manager's characters in approving activities, researcher will use R language; it regularly fetches the data from database, learning the pattern and update the prediction model. It can be interfaced to external Human Resources (HR) module on Management Information System (MIS) to predict the approval status. With minimum effort, this model can be implemented in current MIS-HR that owned by the company to conduct the automatic approval process without manager's interference. Thus, manager can conduct more strategic activities rather than only do the approval process.

Keywords: *Artificial Intelligent, Support Vector Machine, Machine Learning, Management Information System, Manager*

### Abstrak

Perkembangan *Artificial Intelligent* (AI, kecerdasan buatan) dewasa ini semakin maju. Berbagai kemudahan dapat dibuat oleh mesin dengan mempelajari pola khas yang manusia lakukan dan kemudian melakukan ulang pekerjaan tersebut berdasarkan contoh-contoh yang telah diajarkan sebelumnya. Pekerjaan manusia yang telah dipelajari oleh mesin ini dapat dimanfaatkan untuk menggantikan peran manusia dalam melakukan tugas-tugas rutin. Dalam jurnal penelitian ini, penulis memanfaatkan teknologi AI, yaitu *Machine Learning* berbasis *Support Vector Machine* (SVM) yang mampu mempelajari pola khas manajer dalam memberikan keputusan persetujuan, dan kemudian pada gilirannya dipakai sebagai wakil manajer dalam melakukan persetujuan. Dalam mempelajari pola khas manajer memberikan keputusan persetujuan, kode dalam bahasa R akan mengambil data dari *database*, menggunakannya untuk menemukan pola khas persetujuan dan memperbaiki model prediksi yang pada gilirannya akan digunakan Sistem Informasi Manajemen (SIM) Sumberdaya Manusia (SDM) untuk memprediksi persetujuan yang sedang diminta. Dengan perubahan minimal, metode ini dapat diterapkan pada SIM-SDM yang sudah dimiliki perusahaan sebelumnya, sehingga tugas-tugas persetujuan yang sebelumnya dilakukan oleh manajer, akhirnya dapat dilakukan oleh mesin. Dengan demikian, manajer dapat melakukan tugas-tugas yang lebih strategis daripada sekadar melakukan persetujuan rutin.

Kata kunci: *Artificial Intelligent, Support Vector Machine, Machine Learning, Sistem Informasi Manajemen, Manajer*

### 1. Pendahuluan

AI merupakan bidang Ilmu Komputer yang mana mesin dapat dibuat untuk melakukan sesuatu seperti yang dilakukan oleh manusia; dan juga dapat berfikir sesuai dengan yang diinginkan oleh manusia. Di masa depan, diperkirakan semua perangkat elektronika dan komputer menjadi jauh lebih cerdas karena telah ditanamkan berbagai metode AI.

Salah satu model populer dalam AI adalah *Machine Learning* berbasis SVM. Model ini digunakan melakukan prediksi kelas keputusan biner berdasarkan set data latihan yang berisi fitur yang telah disiapkan. SVM akan mempelajari bagaimana menemukan kelompok data berdasarkan bidang paling optimum yang memisahkan antarkelompok. Dengan kemampuan SVM menemukan bidang paling optimum itulah kita dapat

memanfaatkannya untuk mengetahui pola-pola di dalam data. Bila model yang dibangun telah memenuhi tingkat akurasi yang diinginkan, model dapat digunakan untuk memprediksi keputusan berdasarkan pola yang telah ditemukan sebelumnya.

### 2. Metode Penelitian

#### 2.1 Metodologi Penelitian

Penelitian ini menggunakan metode penelitian terapan, di mana hasil penelitian ini dapat diimplementasikan pada bidang-bidang lainnya. Data yang diperoleh berasal dari data primer dan data sekunder. Data primer dilakukan pengumpulan data *log*. Data sekunder diperoleh melalui studi literatur, tulisan ilmiah tentang AI, *Machine Learning*, SVM, SIM-SDM dan tinjauan studi terkait.

## 2.2 Identifikasi Masalah

Pada penelitian ini, peneliti ingin mengetahui apakah SVM dapat menemukan pola khas manajer dalam membuat keputusan persetujuan dan kemudian mampu melakukan memprediksi persetujuan. Juga, peneliti ingin mengetahui pilihan-pilihan lain untuk memperoleh tingkat akurasi prediksi maksimum agar manajer yakin untuk mewakili keputusan selanjutnya kepada mesin.

## 2.3 Rumusan Masalah

Rumusan masalah penelitian ini adalah bagaimana cara model SVM dapat membantu proses SIM-SDM secara lebih baik. Juga, penelitian ini akan mencoba menemukan pilihan optimalisasi SVM agar hasilnya menjadi lebih baik lagi.

## 2.4 Batasan Masalah

Penelitian ini menggunakan bahasa R untuk memprogram model SVM, mengambil data dari *database* mySQL dan perlu dijalankan secara rutin agar modelnya dapat terus-menerus diperbarui. Data penelitian diambil dari suatu perusahaan yang bergerak di bidang perekrutan SDM yang bersifat kontrak dan untuk jenis pekerjaan kerah biru. Data yang diambil berasal dari *log* borang persetujuan "Pemanggilan Pelamar" pada modul SIM-SDM.

## 2.5 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk mengetahui cara penerapan model SVM dalam SIM-SDM untuk menghasilkan sistem yang berdayaguna, serta untuk mengetahui pilihan terbaik untuk meningkatkan akurasi SVM.

## 2.6 Hipotesis

Diduga SVM dapat menemukan pola khas persetujuan manajer dalam kasus pemanggilan pelamar yang sebelumnya dilakukan manajer secara manual di SIM-SDM.

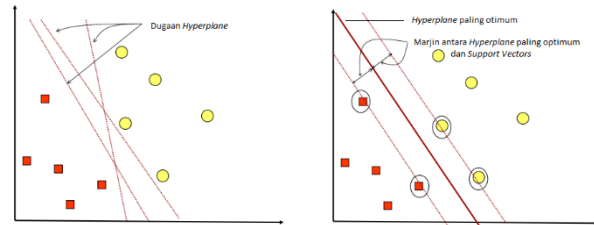
## 2.7 Landasan Teori

### 2.7.1 Support Vector Model (SVM)

Konsep SVM adalah kombinasi teori komputasi yang diinspirasi oleh penelitian Aronszajn tahun 1950, Vapnik tahun 1964 [1], Cover tahun 1965 [2], dan Duda & Hart tahun 1973 [3]; namun belum dirangkai menjadi satu senarai sampai dengan tahun 1992 oleh Boser, Guyon, dan Vapnik yang pertama kali diperkenalkan pada tahun 1992 di ajang *Annual Workshop on Computational* [4].

Pada SVM, data yang ingin dikelompokkan akan dipetakan pada bidang berdimensi tinggi. SVM akan berupaya menemukan bidang pemisah berdimensi lebih rendah dari dimensi awal serta memiliki margin kesalahan paling rendah. Komputer melakukan

pemeriksaan margin kesalahan dari tiap dugaan bidang rekaan agar tahu bidang dugaan mana yang paling optimum memberikan tingkat kesalahan paling kecil. Usaha untuk mencari lokasi bidang berdimensi rendah ini merupakan inti dari proses *Machine Learning* pada SVM [5].



Gambar 1. Cara SVM menemukan bidang pemisah optimum

### 2.7.2 Metode Pengukuran Akurasi

Penelitian ini menggunakan *confusion matrix* untuk mengukur tingkat kebenaran proses klasifikasi SVM. Dengan *confusion matrix* kita dapat menganalisa seberapa baik suatu model dapat mengenali hasil prediksi dari kelas-kelas yang berbeda. Tabel *confusion matrix* ditunjukkan pada tabel berikut:

Tabel 1. Tabel *confusion matrix*

		Prediksi	
		Positif	Negatif
Aktual	Positif	True-Positive (TP)	False-Negative (FN)
	Negatif	False-Positive (FP)	True-Negative (TN)

Akurasi didefinisikan sebagai tingkat kedekatan antara nilai prediksi dengan nilai aktual [6] yang dirumuskan sebagai:

$$Akurasi = \frac{TP + TN}{TP + FN + FP + TN}$$

Dalam penentuan pilihan SVM optimum, peneliti akan membandingkan hasil akurasi pada pilihan bidang/*kernel* sebagai berikut: *rbf*, *poly*, *vanilla*, *tanh*, *laplace*, *bessel*, *anova*, dan *spline*.

### 2.7.3 Sistem Informasi Manajemen (SIM)

SIM merupakan penerapan Sistem Informasi di dalam organisasi untuk mendukung informasi yang dibutuhkan semua tingkatan manajemen. SIM dapat didefinisikan sebagai kumpulan dari interaksi berbagai sistem informasi yang bertanggungjawab mengumpulkan dan mengolah data untuk menyediakan informasi yang berguna untuk semua tingkatan manajemen di dalam kegiatan perencanaan dan pengendalian [7].

Salah satu pemanfaatan SIM adalah di bidang Sumberdaya Manusia (SDM) di mana SIM dipakai untuk pengolahan data personalia [8].

## 2.8 Desain Penelitian

**Tahap 1:** Secara berkala sistem sistem akan mengambil sampel (D) dari *database* dengan mengekstrak rekam jejak persetujuan manajer dan mengambil informasi

yaitu 1). Identitas borang; 2). Identitas manajer; 3). Identitas pelamar; 4). Angka rincian pada borang dengan total 9 data yang diskalakan dari 1 sampai 10; 5). Hasil persetujuan.

**Tahap 2:** Sistem akan melakukan pemeriksaan data, memecah data menjadi set data *training* ( $D_{train}$ ) dan set data *testing* ( $D_{test}$ ).  $D_{train}$  digunakan untuk melaksanakan pelatihan *Machine Learning* SVM dan menghasilkan model-model ( $M_n$ ).  $M$  akan diuji dengan menilai akurasi terhadap beberapa pilihan *kernel* yang nantinya akan dipergunakan lebih lanjut ( $M_{final}$ ). Bila pemeriksaan akurasi terhadap  $D_{test}$  lebih besar dari akurasi yang tersimpan sebelumnya, sistem akan menyimpan  $M_{final}$ .  $M_{final}$  disimpan pada pusat penyimpanan data dan dapat diakses menggunakan *command-line* Sistem Operasi komputer yang digunakan.

**Tahap 3:** Sebuah *file* khusus akan dipersiapkan untuk memproses permintaan dan memberikan prediksi.

**Tahap 4:** SIM-HRD memanggil *file* khusus di tahap 3 untuk mengaplikasikan hasil prediksi sesuai kebutuhan misalnya untuk mengarahkan pilihan *default* di layar aplikasi; dibuatkan *webservice* untuk dipanggil aplikasi SIM-SDM eksternal; atau dapat juga dipakai secara langsung untuk menetapkan persetujuan yang hasilnya langsung disimpan di *database* pada tabel persetujuan.

2.9 Format Data Masukan

Tabel 2. Format data masukan

No	Field	Jenis Data	Keterangan
1	formid	Numerik	ID borang persetujuan. Dalam hal ini borang yang akan dijadikan sampel adalah borang Pemanggilan Pelamar dengan formid bernilai 308
2	mgrid	Numerik	ID manajer yang harus meninjau borang ini
3	applicantid	Numerik	Kode pelamar
4	rating1	Numerik	Skala penilaian pada <i>field</i> 1
5	rating2	Numerik	Skala penilaian pada <i>field</i> 2
6	rating3	Numerik	Skala penilaian pada <i>field</i> 3
7	rating4	Numerik	Skala penilaian pada <i>field</i> 4
8	rating5	Numerik	Skala penilaian pada <i>field</i> 5
9	rating6	Numerik	Skala penilaian pada <i>field</i> 6
10	rating7	Numerik	Skala penilaian pada <i>field</i> 7
11	rating8	Numerik	Skala penilaian pada <i>field</i> 8
12	rating9	Numerik	Skala penilaian pada <i>field</i> 9
13	status	Numerik	Kode berupa: 0 – Pelamar diproses 1 – Pelamar tidak diproses 99 – Data siap ditinjau

3. Hasil dan Pembahasan

3.1 Akses ke Database

Pada bagian ini peneliti akan mengakses *database* *mySQL* sesuai dengan data yang diberikan departemen IT.

Bagian akses ke *database*

```
library(RMariaDB)
db <- dbConnect(RMariaDB::MariaDB(),
  user = 'rstudio',
  password = 'password',
  dbname = 'rstudio',
  host = 'localhost')
sql<-"SELECT * FROM v_appr WHERE formid = 308"
rs <- dbSendQuery(db, sql)
df <- dbRows<-dbFetch(rs)
```

Tabel 3. Penjelasan variabel pada bagian akses *database*

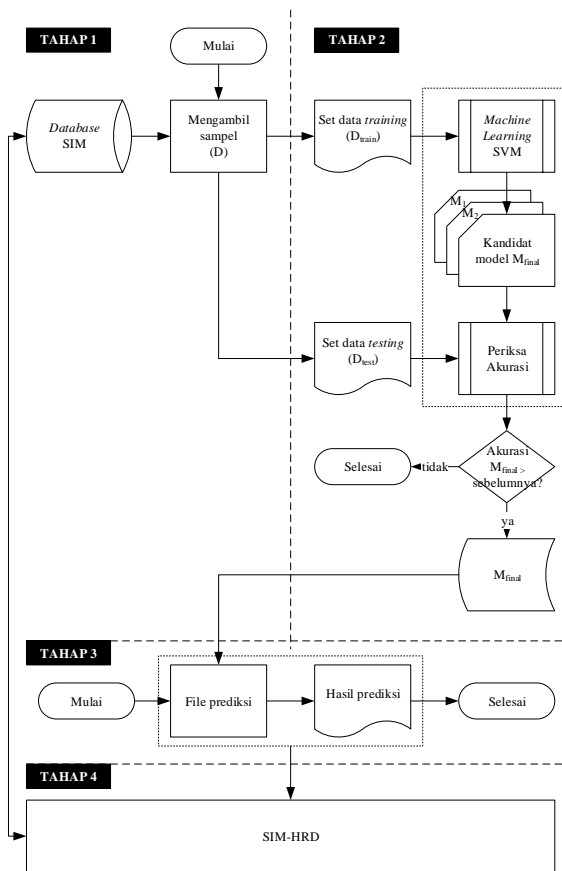
No	Field	Isian	Penjelasan
1	user	rstudio	Nama pengguna yang ditetapkan di server <i>mySQL</i>
2	pass word	pass word	<i>Password</i> yang ditetapkan di server <i>mySQL</i>
3	dbname	rstudio	Nama <i>database</i> yang diberikan oleh departemen TI untuk diakses aplikasi RStudio
4	host	local host	Lokasi <i>server</i> pangkalan data
5	-	v_appr	Nama <i>view</i> atau <i>table</i> yang diberikan departemen TI untuk diakses aplikasi RStudio
6	formid	308	Nomor borang yang dipakai untuk proses penelitian ini

3.2 Pemeriksaan Kandungan Data

Di tahap ini peneliti memeriksa apakah ada data yang kosong.

Bagian pemeriksaan data kosong

```
colsums(is.na(df))
formid mgrid applicantid rating1 rating2
0 0 0 0 0
rating3 rating4 rating5 rating6 rating7
```



Gambar 2. Desain penelitian

```
0 0 0 0 0
rating8 rating9 status
0 0 0
```

Dari sini nampak bahwa semua data terisi baik sehingga tidak diperlukan pembersihan data lebih lanjut. Selain itu, peneliti harus memeriksa apakah ada data *unary* (bernilai tunggal tanpa variasi).

**Bagian pemeriksaan data unary**

```
apply(df, 2, max) - apply(df, 2, min)
formid mgrid applicantid rating1 rating2
0 9 8392718 9 9
rating3 rating4 rating5 rating6 rating7
9 9 9 9 9
rating8 rating9 status
9 9 1
```

Dari sini ditemukan bahwa *formid* adalah data *unary*, dengan demikian kita dapat menghapus variabel ini.

Dari segi bisnis, *mgrid* dan *applicantid* adalah identitas orang yang bersifat unik, tidak memberikan manfaat dalam proses *Machine Learning* selanjutnya sehingga dapat dihapus.

**Bagian hapus variabel yang tidak berguna**

```
df$formid <- NULL
df$applicantid <- NULL
df$mgrid <- NULL
```

**3.3 Pemeriksaan Korelasi Antarvariabel**

Proses selanjutnya adalah memeriksa korelasi antarvariabel yang disediakan (dalam hal ini adalah *rating1* sampai dengan *rating9*) dengan tujuan pembelajarannya (*status*).

**Bagian pemeriksaan korelasi antarvariabel**

```
correlation <- cor(df)
correlation[10, 1:9]
rating1 rating2 rating3 rating4
0.7160014 0.8179037 0.8189337 0.6968002
rating5 rating6 rating7 rating8
0.6827845 0.8189679 0.7566161 0.7122436
rating9
0.4231703
```

Dari data ini terlihat *rating6*, *rating3*, dan *rating2* memiliki nilai korelasi tinggi. Untuk meningkatkan akurasi, pada penelitian ini kita akan memanfaatkan seluruh variabel yang disediakan (tidak dikurangi lagi).

**3.4 Pembagian Set Data**

Data perlu dipecah menjadi set data *training* ( $D_{train}$ ) dan set data *testing* ( $D_{test}$ ) dengan tujuan kita dapat memeriksa akurasi dengan menggunakan data yang berbeda dari proses *Machine Learning*-nya.

**Bagian pembagi set data**

```
library(caTools)
set.seed(NULL)
split <- sample.split(df, splitRatio = 0.8)
train <- subset(df, split == TRUE)
test <- subset(df, split == FALSE)
```

**3.5 Proses Machine Learning SVM**

Di tahap ini peneliti akan melakukan proses *Machine Learning SVM* dengan menjalankan perintah berikut:

**Bagian machine Learning SVM**

```
library(kernlab)
svm <- ksvm(status ~ .,
            data = train,
            type="c-svc",
            kernel = "laplacedot")
```

Pilihan *kernel* ini dapat diubah menjadi berbagai jenis *kernel* lain yang ada misalnya: 1). *rbfdot*; 2). *polydot*; 3). *vanilladot*; 4). *tanhdot*; 5). *laplacedot* (seperti contoh di atas); 6). *besseldot*; 7). *anovadot*; dan juga 8). *splinedot*.

**3.6 Pengukuran Akurasi**

Berikut adalah kode untuk mengukur akurasi model bagi masing-masing pilihan *kernel* yang ada.

**Bagian pengukuran akurasi**

```
prediksi <- predict(svm, newdata = test)
kenyataan <- test$status
library(caret)
confusionMatrix(table(prediksi, kenyataan))
Confusion Matrix and Statistics
```

```
          kenyataan
prediksi 0 1
          0 88 0
          1 2 50
```

```
Accuracy          : 0.9857
P-value [Acc > NIR] : <2e-16
```

Kita perlu mencatat 3 hasil penelitian di tiap *kernel* yaitu 1). Akurasi; 2). Kenyataan salah — prediksi benar; dan 3). Kenyataan benar — prediksi salah.

**3.6 Optimalisasi Pilihan**

Berikut rekapitulasi catatan setiap iterasi proses setiap *kernel Machine Learning SVM* dan catatan akurasinya.

Tabel 4. Rekapitulasi catatan akurasi tiap *kernel*

No.	Jenis Kernel	Akurasi	Kenyataan Salah Prediksi Benar	Kenyataan Benar Prediksi Salah
1	rbfdot	97,86%	3	0
2	polydot	98,57%	1	1
3	vanilladot	98,57%	1	1
4	tanhdot	95,71%	1	5
5	<b>laplacedot</b>	<b>98,57%</b>	<b>2</b>	<b>0</b>
6	besseldot	98,57%	1	1
7	anovadot	98,57%	1	1
8	splinedot	95,00%	2	5

**3.7 Memahami Hasil Machine Learning SVM**

Akurasi terbaik di 98,57% ada pada 5 *kernel* yaitu: *polydot*, *vanilladot*, *laplacedot*, *besseldot*, dan *anovadot*. Secara umum model dikatakan baik karena akurasinya yang tinggi, juga dengan nilai *p-Value* kurang dari 0,05 (dinyatakan signifikan atau hipotesis di bagian 2.6 terpenuhi).

Selain itu, kita amati bahwa *kernel laplacedot* menghasilkan prediksi benar — kenyataan salah sebanyak 2 data. Ini berarti akan ada 2 orang yang terpanggil untuk proses seleksi dan wawancara namun sebetulnya tidak memenuhi standar mutu perusahaan. *Kernel* ini juga menghasilkan prediksi salah — kenyataan benar sebanyak 0 data. Ini berarti tidak ada

orang yang memenuhi standar mutu perusahaan dan tidak terpenggal untuk proses seleksi dan wawancara.

### 3.8 Menyimpan Model

Model dapat disimpan bila akurasinya lebih baik dari proses *Machine Learning SVM* yang dijalankan sebelumnya. Berikut adalah kodenya.

```

Bagian penyimpanan model
akurasi <- 0
load("akurasi.rdata")
km <- table(prediksi, kenyataan)
akurasi.sekarang <- sum(diag(km)) / sum(km)
if(akurasi.sekarang > akurasi) {
  akurasi <- akurasi.sekarang
  save(akurasi, file = "akurasi.rdata")
  save(svm, file = "svm.rdata")
}

```

### 3.8 Menyiapkan Skrip R Lengkap

Untuk membuat skrip R **TAHAP 1** dan **TAHAP 2** pada gambar 2 secara lengkap, kita perlu meletakkan kode R dari bagian 3.1 Akses ke *Database*; 3.2 Pemeriksaan Kandungan Data bagian penghapusan variabel yang tidak berguna; 3.4 Pembagian Set Data; 3.5 Proses *Machine Learning SVM* dengan *kernel 1aplacedot*; 3.6 Pengukuran Akurasi; dan 3.8 Menyimpan Model. Semua bagian tersebut ditulis pada sebuah *file* utuh, misalnya *svm-train.r*.

*File svm-train.r* ini diharapkan untuk dipanggil oleh Sistem Operasi secara berkala (misalnya setiap minggu) agar proses pemuktahiran model dapat dilakukan secara berkesinambungan.

### 3.9 Pembuatan Agen Perwakilan Manajer

Agen perwakilan manajer pada **TAHAP 3** di gambar 2 dibuat dengan mengambil pertanyaan dari argumen pemanggilan aplikasi dengan *field* yang sama dengan ketika proses *Machine Learning SVM* dilakukan. Hasil prediksi ditulis di *console* Sistem Operasi.

```

Bagian agen perwakilan manager
library(kernlab)
load("svm.rdata")
args <- commandArgs(trailingOnly = TRUE)
r1 <- as.numeric(args[1])
r2 <- as.numeric(args[2])
r3 <- as.numeric(args[3])
r4 <- as.numeric(args[4])
r5 <- as.numeric(args[5])
r6 <- as.numeric(args[6])
r7 <- as.numeric(args[7])
r8 <- as.numeric(args[8])
r9 <- as.numeric(args[9])
prediksi <- predict(svm, data.frame(
  rating1 = r1, rating2 = r2,
  rating3 = r3, rating4 = r4,
  rating5 = r5, rating6 = r6,
  rating7 = r7, rating8 = r8,
  rating9 = r9))
cat(prediksi)

```

Bagian ini ditulis pada sebuah *file* terpisah, misalnya *svm-console.r*.

### 3.10 Menguji Pemanggilan Agen Perwakilan Manajer

Untuk menguji prediksi, berikut adalah cara panggilan aplikasi yang dapat dieksekusi langsung di *console*

Sistem Operasi (misalnya pada *command-prompt Windows* atau pada *Terminal* di *Linux/MacOS*). Angka yang dimasukkan adalah 9 data untuk diprediksi dimulai dari *rating1* sampai dengan *rating9*. Hasilnya akan ditunjukkan langsung, apakah 0 (“ditolak”) atau 1 (“dilanjutkan prosesnya”).

```

Contoh pemanggilan svm-console.r
rscript --vanilla svm-console.r 0 0 0 0 0 10 10
10 10 [enter ↵]
1

```

## 4. Kesimpulan dan Saran

### 4.1 Kesimpulan

Dalam penelitian ini, ditemukan bahwa SVM baik dalam mengenali pola khas persetujuan yang dibuat oleh manajer dibuktikan dengan tingkat akurasi yang tinggi.

Dari beberapa pilihan *kernel* yang ada, pilihan *1aplacedot* adalah pilihan *kernel* yang baik karena menghasilkan hanya 1,5% beda harapan persetujuan.

Dengan kondisi ini, jika sistem Pemanggilan Pelamar di SIM-SDM perusahaan ini diubah untuk mengambil prediksi dari SVM, sistem dapat membantu percepatan proses pemanggilan calon tenaga kerja dengan cara mengurangi beban kerja manajer melakukan proses persetujuan; SVM dapat memilih sampai 98,5% calon pelamar bermutu sesuai harapan perusahaan.

### 4.2 Saran

Perusahaan dapat mengubahsui aplikasi SIM-SDM yang digambarkan sebagai **TAHAP 4** pada gambar 2 dengan memanggil agen perwakilan manajer seperti yang diuji pada bagian 3.10 Menguji Pemanggilan Agen Perwakilan Manajer.

Pengembangan lain adalah dengan membuat *webservice* agar hasil *svm-console.r* dapat diakses aplikasi SIM-SDM eksternal.

## Ucapan Terimakasih

Penulis ingin mengucapkan terima kasih kepada Universitas Budi Luhur yang secara keseluruhan telah membantu terlaksananya penelitian ini.

## Daftar Rujukan

- [1] Byun H., Lee S.W., 2003, A Survey on Pattern Recognition Applications of Support Vector Machines, *International Journal of Pattern Recognition and Artificial Intelligence*, 17 (3), pp.459-486
- [2] Tsuda K., 2000, Overview of Support Vector Machine, *Journal of IEICE*, 83 (6), pp.460-466
- [3] Vapnik V.N., 1999, *The Nature of Statistical Learning Theory*, 2nd edition Springer-Verlag, New York Berlin Heidelberg
- [4] Cristianini N., Taylor J.S., 2000, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*, Cambridge Press University
- [5] Nugroho, Anto S., et.al., 2003, *Support Vector Machine, Teori dan Aplikasinya dalam Bioinformatika*, IlmuKomputer.com [Online] tersedia di <http://asnugroho.net/papers/ikcsvm.pdf> [diakses 31 Agustus 2019]

- [6] Sokolova, Marina, et.al, 2009, A systematic analysis of performance measures for classification tasks, *Information Processing & Management*, 45 (4), pp.427-437
- [7] Pangestu, Danu W., 2008, *Teori Dasar Sistem Informasi Manajemen*, IlmuKomputer.com [Online] tersedia di <https://ilmukomputer.org/wp-content/uploads/2008/08/sim.pdf> [diakses 31 Agustus 2019]
- [8] M. Thariq Abdul Aziz, et.al, 2018, Sistem Informasi Manajemen Sumber Daya Manusia Berbasis Web (Studi Kasus: PT. Klik Teknologi Indonesia), *Jurnal TECHNO Nusa Mandiri*, 15 (2), pp.145-146
- [9] Faisal, M. Reza, 2016, *Seri Belajar Pemrograman - Pengenalan Bahasa Pemrograman R*, Indonesia .NET Developer Community
- [10] James Le, 2018, *Support Vector Machines in R*, DataCamp [Online] tersedia di <https://www.datacamp.com/community/tutorials/support-vector-machines-r> [diakses 31 Agustus 2019]