DAFTAR PUSTAKA

[1] Pusat Data dan Informasi Kementrian Kesehatan RI. (2015). Diakses dari :http://www.depkes.go.id/resources/download/pusdatin/infodatin/infodatin-diabetes diakses pada 15 oktober 2018

[2] Dewi, R. P. (2013). *Faktor Risiko Perilaku Yang Berhubungan Dengan Kadar Gula Darah Pada Penderita Diabetes Melitus Tipe 2 di RSUD Kabupaten Karanganyar* (Doctoral Dissertation, Diponegoro University).

[3] Siang, T. C., Soon, T. W., Kasim, S., Mohamad, M. S., Howe, C. W., Deris, S., ... & Ibrahim, Z. (2015). A Review Of Cancer Classification Software For Gene Expression Data. *International Journal Of Bio-Science And Bio-Technology*, 7(4): 89-108.

[4] Zhang, X., Gao, L., Liu, Z. P., & Chen, L. (2015). Identifying Module Biomarker In Type 2 Diabetes Mellitus By Discriminative Area Of Functional Activity. *BMC Bioinformatics*, 16(1): 92.

[5] Chakraborty, D., & Maulik, U. (2014). Identifying Cancer Biomarkers From Microarray Data Using Feature Selection And Semisupervised Learning. *IEEE Journal Of Translational Engineering In Health And Medicine*, 2: 1-11.

[6] Chen, L., Xuan, J., Riggins, R. B., Clarke, R., & Wang, Y. (2011). Identifying Cancer Biomarkers By Network-Constrained Support Vector Machines. *BMC Systems Biology*, 5(1): 161.

[7] Kolberg, J. A., Gerwien, R. W., Watkins, S. M., Wuestehube, L. J., & Urdea, M. (2011). Biomarkers In Type 2 Diabetes: Improving Risk Stratification With The Predx® Diabetes Risk Score. *Expert Review Of Molecular Diagnostics*, 11(8): 775-792.

[8] Mishra, A., Gupta, A., Maheswari, U., & Siddique, L. (2017). Probable Biomarker Identification Using Recursive Feature Extraction And Network Analysis. In *Data Mining Workshops (ICDMW), 2017 IEEE International Conference* On (Pp. 470-477). IEEE.

[9] M. Ni'mah. (2015). *Biomarker Sebagai Molekul Diagnostik Penyakit Kanker*. Dalam halaman https://nanopdf.com/download/biomarker-sebagai-molekul-diagnostik-penyakit-kanker_pdf. diakses pada 28 agustus 2018.

Tahir.(2015). *Biomarker Dan Keutamaannya*. Diakses dari http://repository.unhas.ac.id/bitstream/handle/ diakses pada 4

agustus 2018.

[11] Concepts And Principles Assessment, WHO International Programme On Chemical Safety Biomarkers And Risk. (1993). Diakses dari http://apps.who.int/iris/bitstream/handle/10665/39037/9241571551-eng.pdf?sequence=1&isAllowed=y diakses pada 5 agustus 2018

[12] Fatimah, R. N. (2015). Diabetes Melitus Tipe 2. *Jurnal Majority*, 4(5).

[13] Ndraha, S. (2014). Diabetes Melitus Tipe 2 dan Tatalaksana Terkini. *Departemen Penyakit Dalam Fakultas Kedokteran Universitas Krida Wacana, Jakarta*.

[14] Cahyo, L. B. D. (2018). *Implementasi Metode Support Vector Machine Untuk Melakukan Klasifikasi Pada Data Bioinformatika*. [skripsi].Yogyakarta:Universitas Islam Indonesia

[15] Razavi, A. E. (2012). Dna Microarray, *Isfahan University Of Medical Science*, School Of Pharmacy Department Of Clinical Biochemistry.

[16] Vanitha, C. D. A., Devaraj, D., & Venkatesulu, M. (2015). Gene Expression Data Classification Using Support Vector Machine And Mutual Information-Based Gene Selection. *Procedia Computer Science*, 47: 13-21.

[17] Bolstad, B. M. (2004). *Low-level analysis of high-density oligonucleotide array data: background, normalization and summarization* (Doctoral dissertation, University of California, Berkeley).

[18] Tarca, A. L., Romero, R., & Draghici, S. (2006). Analysis Of Microarray Experiments Of Gene Expression Profiling. *American Journal Of Obstetrics And Gynecology*, 195(2): 373-388.

[19] Irizarry, R. A., Hobbs, B., Collin, F., Beazer‐Barclay, Y. D., Antonellis, K. J., Scherf, U., & Speed, T. P. (2003). Exploration, Normalization, And Summaries Of High Density Oligonucleotide Array Probe Level Data. *Biostatistics*, 4(2): 249-264.

[20] Bourgon, R., Gentleman, R., & Huber, W. (2010). Independent Filtering Increases Detection Power For High-Throughput Experiments. *Proceedings Of The National Academy Of Sciences*, 107(21): 9546-9551.

[21] Heryana, A. (2017). *Uji Chi-Square*. Prodi Kesehatan Masyarakat Fikes Univeritas Esa Unggul.

[22] Silalahi, D. K., Murfi, H., & Satria, Y. (2017). Studi Perbandingan Pemilihan Fitur Untuk Support Vector Machine Pada Klasifikasi Penilaian Risiko Kredit. *Edumatsains*, 1(2): 119-136.

B. (2013). *Information Gain Feature Selection Based On Feature*

Interactions (Doctoral Dissertation).

[24] Shannon, C. E. (1948). A Mathematical Theory Of Communication. *Bell System Technical Journal*, 27(3): 379-423.

[25] Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1): 5-32.

[26] Louppe, G., Wehenkel, L., Sutera, A., & Geurts, P. (2013). Understanding Variable Importances In Forests Of Randomized Trees. In *Advances In Neural Information Processing Systems* (Pp. 431-439).

[27] Tharwat, A., Gaber, T., Ibrahim, A., & Hassanien, A. E. (2017). Linear Discriminant Analysis: A Detailed Tutorial. *AI Communications*, 30(2): 169-190

[28] Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification And Regression Trees* (Monterey, California: Wadsworth).

[29] Siahaan, D., Wahyuningsih, S., & Amijaya, F. D. T. (2017). Aplikasi Classification And Regression Tree (CART) Dan Regresi Logistik Ordinal Dalam Bidang Pendididikan. *Jurnal Eksponensial*, 7(1): 95-104.

[30] Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene Selection For Cancer Classification Using Support Vector Machines. *Machine Learning*, 46(1-3): 389-422.

[31] Obuchowski, N. A. (2003). Receiver operating characteristic curves and their use in radiology. *Radiology*, 229(1): 3-8.

[32] Park, S. H., Goo, J. M., & Jo, C. H. (2004). Receiver operating characteristic (ROC) curve: practical review for radiologists. *Korean Journal of Radiology*, 5(1):11-18.

[33] Brefeld, U., & Scheffer, T. (2005). AUC maximizing support vector learning. In *Proceedings of the ICML 2005 workshop on ROC Analysis in Machine Learning.*

LAMPIRAN

Lampiran 1 Package yang digunakan

```
library(GEOquery)
library(Biobase)
library(affy)
library(genefilter)
library(gcrma)
library(stats4)
library(IRanges)
library(S4Vectors)
library(org.Hs.eg.db)
library(AnnotationDbi)
library(hgu133plus2.db)
library(hgu133acdf)
library(hgu133plus2cdf)
library(hgu133a.db)
library(simpleaffy)
library(preprocessCore)
library(affyPLM)
library(xlsx)
library(FSelector)
library(lattice)
library(ggplot2)
library(caret)
library(e1071)
library(pROC)
library(PRROC)
library(ROSE)
library(randomForest)
```

n 2 Preprosesing

```
a<- threestep(gse18732, background.method = "RMA.2",
ze.method="quantile",summary.method="median.polish")
_data)
```

Lampiran 3 Filtering

```
filter <- nsFilter(rma_data, require.entrez =T, remove.dupEntrez =
T,var.cutoff = 0.5, feature.exclude = "^AFFX")

log <-filter$filter.log

eset <- filter$eset

featureNames(eset) <- make.names(featureNames(eset))

View(eset)

dim(eset)

rawdata.exprs = t(exprs(eset))

norm_rawdata2=as.data.frame(rawdata.exprs)

norm_rawdata2$y=label
```

Lampiran 4 Chi-Square Ranking

```
weights <- chi.squared(y~.,norm_rawdata2[,1:10094])

subset.CSQR<- cutoff.k(weights, 500)

#Gene name dan gene ontology

id.CSQRgene = as.character(NULL)

for(i in 1:500){

  id.CSQRgene[i] <- substring(subset.CSQR[i],2)

}

select(hgu133plus2.db, id.CSQRgene, "GENENAME")

gennameCSQR <- select(hgu133plus2.db, id.CSQRgene,
c("SYMBOL","GENENAME"), "PROBEID")

ontologyCSQR <- select(hgu133plus2.db, id.CSQRgene,
c("SYMBOL","GENENAME","ENTREZID","ONTOLOGY"), "PROBEID")

top500 <- NULL

for(i in 1:500){

  top500[i] <- grep(subset.CSQR[i],colnames(norm_rawdata2))

}

norm_rawdata3<- norm_rawdata2[,top500]

wdata3$y=label
```

n 5 Information Gain Ranking

```
.gain <- information.gain(y~., norm_rawdata3)
```

```
subset.gain<- cutoff.k(weights.gain, 400)#Gene name dan gene
ontology

id.IGgene = as.character(NULL)

for(i in 1:400){

  id.IGgene[i] <- substring(subset.gain[i],2)

}

select(hgu133plus2.db, id.IGgene, "GENENAME")

gennameIG <- select(hgu133plus2.db, id.IGgene,
c("SYMBOL","GENENAME"), "PROBEID")

ontologyIG <- select(hgu133plus2.db, id.IGgene,
c("SYMBOL","GENENAME","ENTREZID","ONTOLOGY"), "PROBEID")

top400 <- NULL

for(i in 1:400){

  top400[i] <- grep(subset.gain[i],colnames(norm_rawdata3))

}

norm_rawdata4<- norm_rawdata3[,top400]

norm_rawdata4$y=label
```

Lampiran 6 Random Forest Importance Ranking

```
weights.rf <- random.forest.importance(y~., norm_rawdata4,
importance.type = 1)

subset.rf<- cutoff.k(weights.rf, 300)

id.RFIgene = as.character(NULL)

for(i in 1:300){

  id.RFIgene[i] <- substring(subset.rf[i],2)

}

select(hgu133plus2.db, id.RFIgene, "GENENAME")

gennameRFI <- select(hgu133plus2.db, id.RFIgene,
c("SYMBOL","GENENAME"), "PROBEID")

ontologyRFI <- select(hgu133plus2.db, id.RFIgene,
c("SYMBOL","GENENAME","ENTREZID","ONTOLOGY"), "PROBEID")

top300 <- NULL

    1:300){

    [i] <- grep(subset.rf[i],colnames(norm_rawdata4))
```

```
norm_rawdata5<- norm_rawdata4[,top300]
norm_rawdata5$y=label
```

Lampiran 7 Seleksi Fitur LDA

```
best.acc <- 0
acc <- NULL
pb <- txtProgressBar(min = 0, max = 100, style = 3)

for(i in 1:100){
index = createDataPartition(y=norm_rawdata5$y, p=0.7, list=FALSE)
train = norm_rawdata5[index,]
test = norm_rawdata5[-index,]
tes.label <- test$y
lda.fit = train(y ~ ., data=train, method="lda",
                    trControl = trainControl(method = "cv",number =
10))
print(lda.fit)
summary(lda.fit)
pred.y = predict(lda.fit, test)
matrixConfussion <- confusionMatrix(tes.label,pred.y)
acc[i]<- matrixConfussion[["overall"]][["Accuracy"]]
if(acc[i]>best.acc){
  best.acc<-acc[i]
  best.matrixCon <- matrixConfussion
  best.train <- train
  best.model <- lda.fit
}
setTxtProgressBar(pb,i)
}
close(pb)
      <- predictors(lda.fit)
      ame dan gene ontology
      ene = as.character(NULL)
```

```
for(i in 1:300){
  id.LDAgene[i] <- substring(geneLDA[i],2)
}
select(hgu133plus2.db, id.LDAgene, "GENENAME")
gennameLDA <- select(hgu133plus2.db, id.LDAgene,
c("SYMBOL","GENENAME"), "PROBEID")
ontologyLDA <- select(hgu133plus2.db, id.LDAgene,
c("SYMBOL","GENENAME","ENTREZID","ONTOLOGY"), "PROBEID")
```

Lampiran 8 Seleksi Fitur RandomForest RFE

```
best.acc1 <- 0
acc1 <- NULL
pb <- txtProgressBar(min = 0, max = 100, style = 3)
control <- rfeControl(functions=rfFuncs, method="cv", number=10)
# run the RFE algorithm
results <- rfe(norm_rawdata5[,1:300], norm_rawdata5[,301],
sizes=c(1:300), rfeControl=control)
# summarize the results
print(results)
# list the chosen features
geneRF <- predictors(results)
#Gene name dan gene ontology
id.RFgene = as.character(NULL)
for(i in 1:300){
  id.RFgene[i] <- substring(geneRF[i],2)
}
select(hgu133plus2.db, id.RFgene, "GENENAME")
gennameRF <- select(hgu133plus2.db, id.RFgene,
c("SYMBOL","GENENAME"), "PROBEID")
ontologyRF <- select(hgu133plus2.db, id.RFgene,
c("SYMBOL","GENENAME","ENTREZID","ONTOLOGY"), "PROBEID")
```

n 9 Seleksi Fitur SVM

```
cSVM <- 0
 <- NULL
```

```
pb.SVM <- txtProgressBar(min = 0, max = 100, style = 3)

for(i in 1:100){

indexSVM = createDataPartition(y=norm_rawdata5$y, p=0.7,
list=FALSE)

trainSVM = norm_rawdata5[index,]

testSVM = norm_rawdata5[-index,]

tes.labelSVM <- test$y

svm_Radial <- train(y ~., data = trainSVM, method = "svmRadial",

                    trControl=trainControl(method = "cv",number =
10),

                    tuneLength = 10)

test_pred_Radial <- predict(svm_Radial, newdata = testSVM)

matrixconfussionSVM <-
confusionMatrix(tes.labelSVM,test_pred_Radial )

acc.SVM[i]<- matrikxconfussionSVM[["overall"]][["Accuracy"]]

if(acc.SVM[i]>best.accSVM){

  best.accSVM<-acc.SVM[i]

  best.matrixConSVM <- matrixconfussionSVM

  best.trainSVM <- trainSVM

  best.modelSVM <- svm_Radial

}

setTxtProgressBar(pb.SVM,i)

}

close(pb.SVM)

geneSVM <- predictors(svm_Radial)

#Gene name dan gene ontology

id.SVMgene = as.character(NULL)

for(i in 1:300){

  id.SVMgene[i] <- substring(geneSVM[i],2)

}

     ngu133plus2.db, id.SVMgene, "GENENAME")

     SVM <- select(hgu133plus2.db, id.SVMgene,
     DL","GENENAME"), "PROBEID")
```

```
ontologySVM <- select(hgu133plus2.db, id.SVMgene,
c("SYMBOL","GENENAME","ENTREZID","ONTOLOGY"), "PROBEID")
```

Lampiran 10 Tabel Perbandingan Hasil

```
best.LDA70acc<-
rbind(best.acc52,best.acc10,best.acc20,best.acc30,best.acc40,best.
acc50,best.acc100,best.acc150,best.acc200,best.acc)

best.SVM70acc<-
rbind(best.accSVM5.70,best.accSVM10.70,best.accSVM20.70,best.accSV
M30.70,best.accSVM40.70,best.accSVM50.70,best.accSVM100.70,best.ac
cSVM150.70,best.accSVM200.70,best.accSVM)

best.RF70acc<-
rbind(best.acc.rf5.70,best.acc.rf10.70,best.acc.rf20.70,best.acc.r
f30.70,best.acc.rf40.70,best.acc.rf50.70,best.acc.rf100.70,best.ac
c.rf150.70,best.acc.rf200.70,best.acc.rf300.70)

best.SVM70.auc<-
cbind(best.SVM70acc,c(roc.plot5.70$auc,roc.plot10.70$auc,roc.plot2
0.70$auc,roc.plot30.70$auc,roc.plot40.70$auc,roc.plot50.70$auc,roc
.plot100.70$auc,roc.plot150.70$auc,roc.plot200.70$auc,roc.plot.70$
auc))

best.LDA70.auc<-
cbind(best.LDA70acc,c(roc.plotLDA5.70$auc,roc.plotLDA10.70$auc,roc
.plotLDA20.70$auc,roc.plotLDA30.70$auc,roc.plotLDA40.70$auc,roc.pl
otLDA50.70$auc,roc.plotLDA100.70$auc,roc.plotLDA150.70$auc,roc.plo
tLDA200.70$auc,roc.plotLDA.70$auc))

best.RF70.auc<-
cbind(best.RF70acc,c(roc.plotrf5.70$auc,roc.plotrf10.70$auc,roc.pl
otrf20.70$auc,roc.plotrf30.70$auc,roc.plotrf40.70$auc,roc.plotrf50
.70$auc,roc.plotrf100.70$auc,roc.plotrf150.70$auc,roc.plotrf200.70
$auc,roc.plotrf300.70$auc))

best.LDA70<-
cbind(best.LDA70.auc,c(best.matrixCon52[["byClass"]][["Sensitivity
"]],best.matrixCon10[["byClass"]][["Sensitivity"]],best.matrixCon2
0[["byClass"]][["Sensitivity"]],best.matrixCon30[["byClass"]][["Se
nsitivity"]],best.matrixCon40[["byClass"]][["Sensitivity"]],best.m
atrixCon50[["byClass"]][["Sensitivity"]],best.matrixCon100[["byCla
ss"]][["Sensitivity"]],best.matrixCon150[["byClass"]][["Sensitivit
y"]],best.matrixCon200[["byClass"]][["Sensitivity"]],best.matrixCo
n[["byClass"]][["Sensitivity"]]))

best.LDA70<-
cbind(best.LDA70.auc,c(best.matrixCon52[["byClass"]][["Sensitivity
"]],best.matrixCon10[["byClass"]][["Sensitivity"]],best.matrixCon2
lass"]][["Sensitivity"]],best.matrixCon30[["byClass"]][["Se
ty"]],best.matrixCon40[["byClass"]][["Sensitivity"]],best.m
n50[["byClass"]][["Sensitivity"]],best.matrixCon100[["byCla
'Sensitivity"]],best.matrixCon150[["byClass"]][["Sensitivit
```

```
y"]],best.matrixCon200[["byClass"]][["Sensitivity"]],best.matrixCo
n[["byClass"]][["Sensitivity"]]))

best.LDA70<-
cbind(best.LDA70,c(best.matrixCon52[["byClass"]][["Specificity"]],
best.matrixCon10[["byClass"]][["Specificity"]],best.matrixCon20[["
byClass"]][["Specificity"]],best.matrixCon30[["byClass"]][["Specif
icity"]],best.matrixCon40[["byClass"]][["Specificity"]],best.matri
xCon50[["byClass"]][["Specificity"]],best.matrixCon100[["byClass"]
][["Specificity"]],best.matrixCon150[["byClass"]][["Specificity"]]
,best.matrixCon200[["byClass"]][["Specificity"]],best.matrixCon[["
byClass"]][["Specificity"]]))

best.SVM70<-
cbind(best.SVM70.auc,c(best.matrixConSVM5.70[["byClass"]][["Sensit
ivity"]],best.matrixConSVM10.70[["byClass"]][["Sensitivity"]],best
.matrixConSVM20.70[["byClass"]][["Sensitivity"]],best.matrixConSVM
30.70[["byClass"]][["Sensitivity"]],best.matrixConSVM40.70[["byCla
ss"]][["Sensitivity"]],best.matrixConSVM50.70[["byClass"]][["Sensi
tivity"]],best.matrixConSVM100.70[["byClass"]][["Sensitivity"]],be
st.matrixConSVM150.70[["byClass"]][["Sensitivity"]],best.matrixCon
SVM200.70[["byClass"]][["Sensitivity"]],best.matrixConSVM.70[["byC
lass"]][["Sensitivity"]]))

best.SVM70<-
cbind(best.SVM70,c(best.matrixConSVM5.70[["byClass"]][["Specificit
y"]],best.matrixConSVM10.70[["byClass"]][["Specificity"]],best.mat
rixConSVM20.70[["byClass"]][["Specificity"]],best.matrixConSVM30.7
0[["byClass"]][["Specificity"]],best.matrixConSVM40.70[["byClass"]
][["Specificity"]],best.matrixConSVM50.70[["byClass"]][["Specifici
ty"]],best.matrixConSVM100.70[["byClass"]][["Specificity"]],best.m
atrixConSVM150.70[["byClass"]][["Specificity"]],best.matrixConSVM2
00.70[["byClass"]][["Specificity"]],best.matrixConSVM.70[["byClass
"]][["Specificity"]]))

best.RF70<-
cbind(best.RF70.auc,c(best.matrixCon.rf5.70[["byClass"]][["Sensiti
vity"]],best.matrixCon.rf10.70[["byClass"]][["Sensitivity"]],best.
matrixCon.rf20.70[["byClass"]][["Sensitivity"]],best.matrixCon.rf3
0.70[["byClass"]][["Sensitivity"]],best.matrixCon.rf40.70[["byClas
s"]][["Sensitivity"]],best.matrixCon.rf50.70[["byClass"]][["Sensit
ivity"]],best.matrixCon.rf100.70[["byClass"]][["Sensitivity"]],bes
t.matrixCon.rf150.70[["byClass"]][["Sensitivity"]],best.matrixCon.
rf200.70[["byClass"]][["Sensitivity"]],best.matrixCon.rf300.70[["b
yClass"]][["Sensitivity"]]))

best.RF70<-
cbind(best.RF70,c(best.matrixCon.rf5.70[["byClass"]][["Specificity
"]],best.matrixCon.rf10.70[["byClass"]][["Specificity"]],best.matr
ix[["byClass"]][["Specificity"]],best.matrixCon.rf30.70
ass"]][["Specificity"]],best.matrixCon.rf40.70[["byClass"]]
ficity"]],best.matrixCon.rf50.70[["byClass"]][["Specificit
st.matrixCon.rf100.70[["byClass"]][["Specificity"]],best.ma
.rf150.70[["byClass"]][["Specificity"]],best.matrixCon.rf20
```

```
0.70[["byClass"]][["Specificity"]],best.matrixCon.rf300.70[["byCla
ss"]][["Specificity"]]))

dimnames(best.LDA70) =
list(c("5","10","20","30","40","50","100","150","200","300"),c("Ac
curacy(%)","AUC","Sensitivity","Specificity"))

dimnames(best.SVM70) =
list(c("5","10","20","30","40","50","100","150","200","300"),c("Ac
curacy(%)","AUC","Sensitivity","Specificity"))

dimnames(best.RF70) =
list(c("5","10","20","30","40","50","100","150","200","300"),c("Ac
curacy(%)","AUC","Sensitivity","Specificity"))
```

Lampiran 11 Nama dan Simbol Gen

GEN ID	SIMBOL GEN	NAMA GEN
206896_s_at	GNG7	G protein subunit gamma 7
1554628_at	ZNF57	zinc finger protein 57
1561539_at	LOC100506368	uncharacterized LOC100506368
227203_at	FBXL17	F-box and leucine rich repeat protein 17
200650_s_at	LDHA	lactate dehydrogenase A
206030_at	ASPA	aspartoacylase
227812_at	TNFRSF19	TNF receptor superfamily member 19
202611_s_at	MED14	mediator complex subunit 14
205145_s_at	MYL5	myosin light chain 5