

DAFTAR PUSTAKA

- Alfiansyah, B, (2008), Diskritisasi Variabel *Credit Scoring* Model menggunakan Metode Entropi dan Khi Kuadrat,[Skripsi], Statistika FMIPA IPB,
- Andana, A, P., Diah S., dan Agus R, (2017), *Model Regresi Menggunakan Least Absolute Shrinkage And Selection Operator (LASSO) Pada Data Banyaknya Gizi Buruk Kabupaten/Kota Di Jawa Tengah*, Jurnal Gaussian, Vol, 6, No, 1,
- Ardita, D, I, (2014), *Perbandingan Metode Diskritisasi Dalam Model Regresi Logistik (Studi Kasus: Pembentukan Model Penskoran Kredit Bank X)*,
- Aziz, F, (2020), *Peningkatan performance Logistic Regression menggunakan teknik Ensemble Bagging pada kasus Credit Scoring*, Vol, 1, No,1, 21–27,
- Bank Indoneisa, 2021, Survei Perbankan Triwulan IV-2021,
- Fatmasari, N, (2013), *Analisis Sistem Pembiayaan Kpr Bank Konvensional Dan Pembiayaan Kprs Bank Syariah (Studi Kasus Bank Btn Dengan Bank Muamalat)*, Jurnal Akuntansi, Vol,1, No,3
- Hakim, A., Mukid, M,A., Yasin, H., dan Sugito, (2019), *Analisis Klasifikasi Credit Scoring Menggunakan Weighted Probabilistic Neural Network (WPNN)*, Jurnal Statistika, Vol, 7 No, 1,
- Hastie, T., Tibshirani, R, dan Friedman, J, (2008), *The Elements of Statistical Learning, Data Mining, Inference, and Prediction, Second Edition*, Springer, New York,
- Hollowel, (2004), *A fair Isaac white paper : Technology Guide To The Scorecard Module*,
- Kantardzic, M, (2003), *Data Mining : Concepts, Models, Methods, And Algorithms*, IEEE and Wiley Inter-Science, New York,

- Nurwahidah, A, I., Abdul K., dan Suliadi, (2015), *Model Scredit Scoring Menggunakan Regresi Logistik Beserta Validasinya*, Prosiding KNMSA
- Siddiqi, N, 2006, *Credit Risk Scorecard Developing and Implementing Intelligent Credit Scoring*, New Jersey (US) : John Willey & Sons,
- Tabagari S, 2015, *Credit Scoring by Logistic Regression* [tesis], Tartu (EE): University of Tartu,
- Tampil, Y, A., Komalig, H., dan Langi, Y, (2015), *Logistic Regression Analysis To Determine Factors Affecting The Grade Point Average (GPA) Of FMIPA Student Of Sam Ratulangi University Of Manado*, JdC, Vol, 6, No,2, 57-62
- Tibshirani R, (1996), *Regression shrinkage and selection via the LASSO*, *Journal of the Royal Statistical Society*, Vol, 58, No,1

LAMPIRAN

Lampiran 1. Data variabel respon dan variabel prediktor

Status	Age	Jobtype	Jenis Agunan	Jumlah Pinjaman	Jangka Waktu Pinjam	Current InterestRate	Tanggungan
Bad	40	Employee	Rumah Tinggal	141000000	9	0,1425	2
Good	37	Employee	Rumah Tinggal	400000000	16	0,1425	3
Good	32	Employee	Rumah Tinggal	150000000	10	0,1325	2
Bad	37	Employee	Rumah Tinggal	300000000	15	0,1325	3
Bad	35	Employee	Rumah Tinggal	313000000	13	0,1325	5
Bad	41	Employee	Rumah Tinggal	354000000	14	0,1325	4
Good	34	Self Employee	Rumah Tinggal	796000000	10	0,1325	3
Good	28	Employee	Rumah Tinggal	155000000	10	0,1325	2
Good	25	Employee	Rumah Tinggal	199824000	15	0,1325	3
Bad	32	Employee	Rumah Tinggal	163000000	15	0,1325	4
Bad	41	Employee	Rumah Tinggal	207000000	10	0,1325	3
Bad	34	Employee	Rumah Tinggal	301000000	20	0,1325	2
Good	25	Employee	Rumah Tinggal	220000000	15	0,1325	1
Bad	32	Employee	Apartement	331063000	10	0,1325	5
Good	31	Employee	Apartement	349600000	10	0,1325	0
Good	28	Profesional	Rumah Tinggal	397395000	15	0,1325	3
Good	25	Employee	Rumah Tinggal	467000000	15	0,1325	4
Good	31	Profesional	Rumah Tinggal	692300000	15	0,1325	2
Bad	39	Employee	Rumah Tinggal	1491210000	15	0,1325	5
:	:	:	:	:	:	:	:
Good	31	Employee	Rumah Tinggal	384000000	15	0,1325	3
Good	36	Employee	Rumah Tinggal	189000000	15	0,1325	3
Bad	25	Self Employee	Apartement	259200000	10	0,1325	0
Good	38	Employee	Rumah Tinggal	336000000	12	0,1325	3
Bad	40	Employee	Rumah Tinggal	411958000	15	0,1325	4

Lampiran 2. Syntax menggunakan R studio

```
library(readxl)
topup_mandiri <- read_excel("D:/S-WORD/Data/toupmandiri.xlsx")

#Weight of Evidence
library(InformationValue)

#WOE variabel
age
WOE_age <- WOE(topup_mandiri$Agegroup,as.numeric(topup_mandiri$Status))
Tabel_WOE_age <- WOETable(topup_mandiri$Agegroup,
                           as.numeric(topup_mandiri$Status))
Tabel_WOE_age

#WOE variabel jobtype
WOE_pekerjaan <- WOE(topup_mandiri$Jobtype,
                     as.numeric(topup_mandiri$Status))
Tabel_WOE_pekerjaan <- WOETable(topup_mandiri$Jobtype,
                                 as.numeric(topup_mandiri$Status))
Tabel_WOE_pekerjaan

#WOE variabel jenis agunan
WOE_agunan <- WOE(topup_mandiri
                  $JenisAgunan,as.numeric(topup_mandiri$Status))
Tabel_WOE_agunan <- WOETable(topup_mandiri$JenisAgunan,
                              as.numeric(topup_mandiri$Status))
Tabel_WOE_agunan

#WOE variabel pinjaman
WOE_pinjaman<-WOE(topup_mandiri$pinjamangroup,
```

```

as.numeric(topup_mandiri$Status))
Tabel_WOE_pinjaman <- WOETable(topup_mandiri$pinjamangroup,
as.numeric(topup_mandiri$Status))
Tabel_WOE_pinjaman

#WOE variabel jangka waktu pinjaman
WOE_jwaktu <- WOE(topup_mandiri$jwaktugroup,
as.numeric(topup_mandiri$Status))
Tabel_WOE_jwaktu <- WOETable(topup_mandiri$jwaktugroup,
as.numeric(topup_mandiri$Status))
Tabel_WOE_jwaktu

#WOE variabel suku bunga
WOE_sukubunga <- WOE(topup_mandiri$sukubungagroup,
as.numeric(topup_mandiri$Status))
Tabel_WOE_sukubunga <- WOETable(topup_mandiri$sukubungagroup,
as.numeric(topup_mandiri$Status))
Tabel_WOE_sukubunga

#WOE variabel tanggungan
WOE_tanggungan <- WOE(topup_mandiri$tanggunganangroup,
as.numeric(topup_mandiri$Status))
Tabel_WOE_tanggungan <- WOETable(topup_mandiri$tanggunganangroup,
as.numeric(topup_mandiri$Status))
Tabel_WOE_tanggungan

#10 -fold cross-validation
cv_model <- cv.glmnet(x_train, y_train, alpha = 0.1)
cv_model
best_lambda <- cv_model$lambda.min

#display optimal lambda value
best_lambda

```

```

#view plot of test MSE's vs. lambda values
plot(cv_model)

plot(cv_model$lambda)

#view coefficients of best model lasso fit
lassofit <- glmnet(x_train, y_train, alpha = 1, lambda = best_lambda)
coef(lassofit)

#predicted values
y_predicted <- predict(lassofit, s = best_lambda, newx = x_test)

library(caret)
y_predicted = as.data.frame(y_predicted)
ytrain = as.data.frame(y_train)
y_predicted

y_predicted$prediksi = ifelse(y_predicted$s1 > 0.5, 1, 0 )
y_predicted$aktual = factor(ifelse(y_test == 1, "good", "bad"))
str(y_predicted)

tab1 <- table(Predicted=y_predicted$prediksi, Actual=y_predicted$aktual)
tab1

akurasi <- ((tab1[1,1]+tab1[2,2])/sum(tab1)*100)
akurasi

#Score Scaling
tabel_WOE <- rbind(
  cbind(input="WOE Age", Tabel_WOE_age [,c(1,7)]),
  cbind(input="WOE Pekerjaan", Tabel_WOE_pekerjaan [,c(1,7)]),

```

```

cbind(input="WOE Agunan", Tabel_WOE_agunan[,c(1,7)]),
cbind(input="WOE Jumlah Pinjaman", Tabel_WOE_pinjaman[,c(1,7)]),
cbind(input="WOE Jangka Waktu Pinjaman", Tabel_WOE_jwaktu[,c(1,7)]),
cbind(input="WOE Suku Bunga", Tabel_WOE_sukubunga[,c(1,7)]),
cbind(input="WOE Tanggungan", Tabel_WOE_tanggungan[,c(1,7)])

```

```

factor_value <- 20/log(2)
factor_value

```

```

offset_value <- 600-(factor_value*log(20))
offset_value

```

```

offset_value <- 600-factor_value*log(50)
offset_value

```

```

beta <- coef(lassofit)
beta

```

```

betawoe<-rep(beta[2:8],times=as.vector (table(tabel_WOE[,1])))

```

```

tabel_WOE[,4] <- round((((betawoe*tabel_WOE[,3] + beta[1]/7)) * factor_value +
  offset_value/7)

```

```

colnames(tabel_WOE)[4] <- "score"
tabel_WOE

```