電信公司顧客流失預測

第五組

Agenda

01 研究背景/資料集介紹

02 變量分析

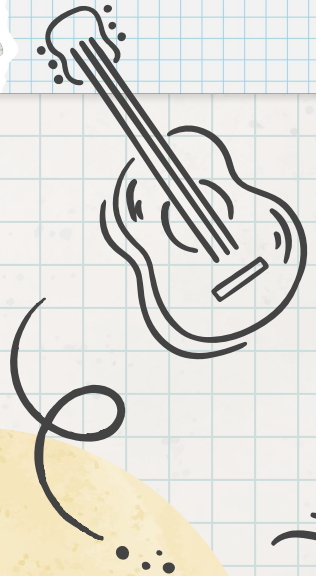
03 Random Forest

04 xgboost

05 Logistic Regression

06 neural network

07 分析與總結



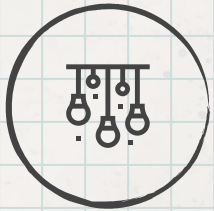
01

研究背景與資料介紹

研究背景

- 研究顧客的流失行為是每間公司的重要議題。藉由資料分析可挖掘出促使顧客流失的關鍵要素，使公司能夠著手改善之。本次研究以電信公司為例，分析資料並建模預測。

資料及介紹

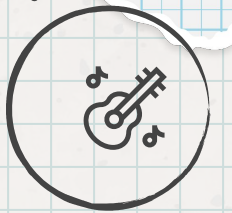


資料來源

IBM/Kaggle

樣本數

7043筆



欄位

21欄

資料來源

欄位名稱	欄位說明	變數類型
customerID	顧客的個人ID	連續型
gender	顧客的性別	類別型
SeniorCitizen	顧客是否是老年人	類別型
Partner	顧客是否有伴侶	類別型
Dependents	顧客是否有家屬	類別型
tenure	顧客已經簽約多少時間了	連續型
PhoneService	顧客是否有手機服務	類別型
MultipleLines	顧客是否有多個門號	類別型
InternetService	顧客是否有網路服務	類別型
OnlineSecurity	顧客是否有網路安全服務	類別型
OnlineBackup	顧客是否有線上備份服務	類別型

欄位名稱	欄位說明	變數類型
DeviceProtection	顧客是否有裝置保護服務	類別型
TechSupport	顧客是否有科技支援	類別型
StreamingTV	顧客是否有串流電視	類別型
StreamingMovies	顧客是否有串流電影	類別型
Contract	顧客的合約類型	類別型
PaperlessBilling	顧客是否是用無紙本帳單	類別型
PaymentMethod	顧客的繳費方式	類別型
MonthlyCharges	顧客每個月的費用	連續型
TotalCharges	顧客每季的總帳單	連續型
Churn	顧客是否流失	類別型

資料前處理

na.omit

刪除一些缺失值

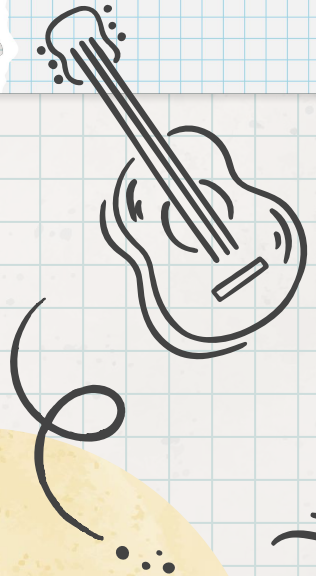
rescale numerical variable

把連續型變數 rescale

resampling

10-fold cross validation

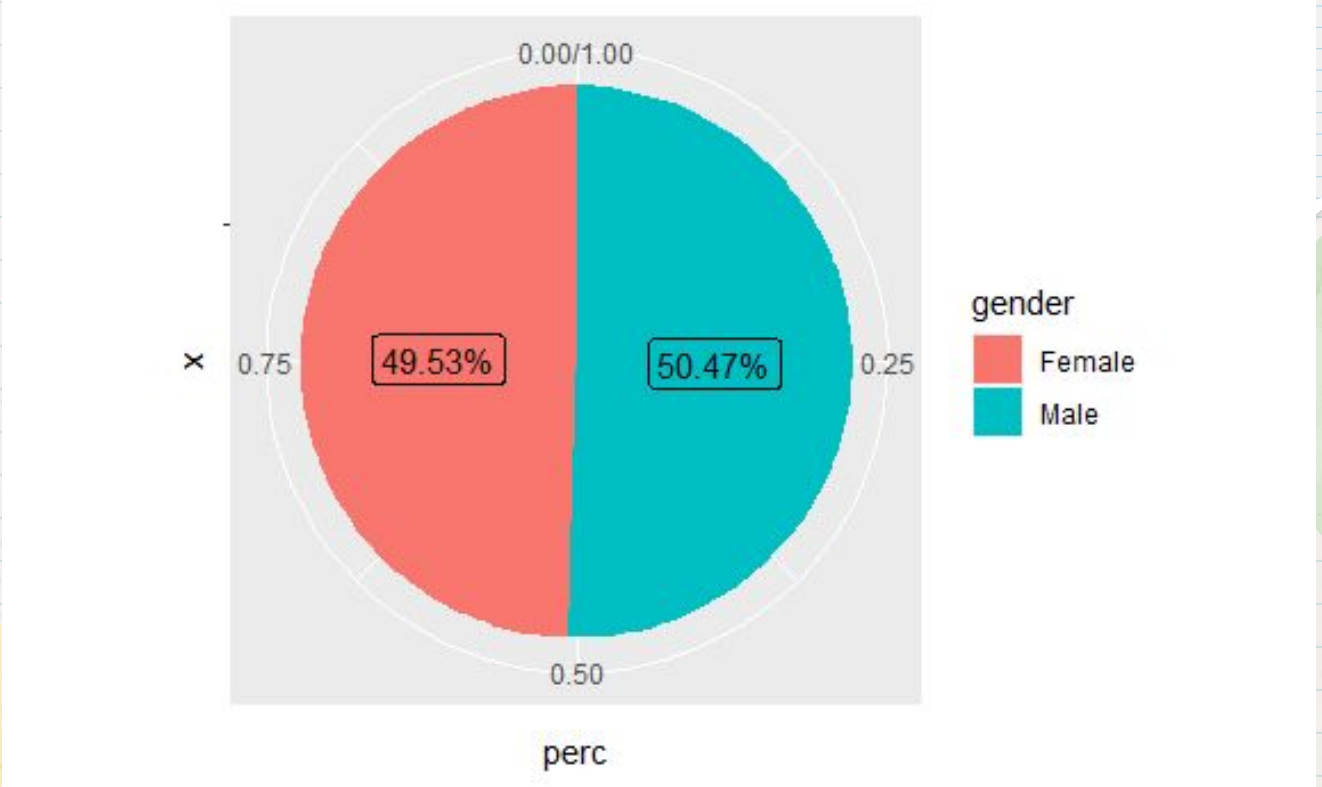
```
#10-fold cross validation
trControl <- trainControl(method = 'cv',
                          number = 10,
                          repeats =  1)
```



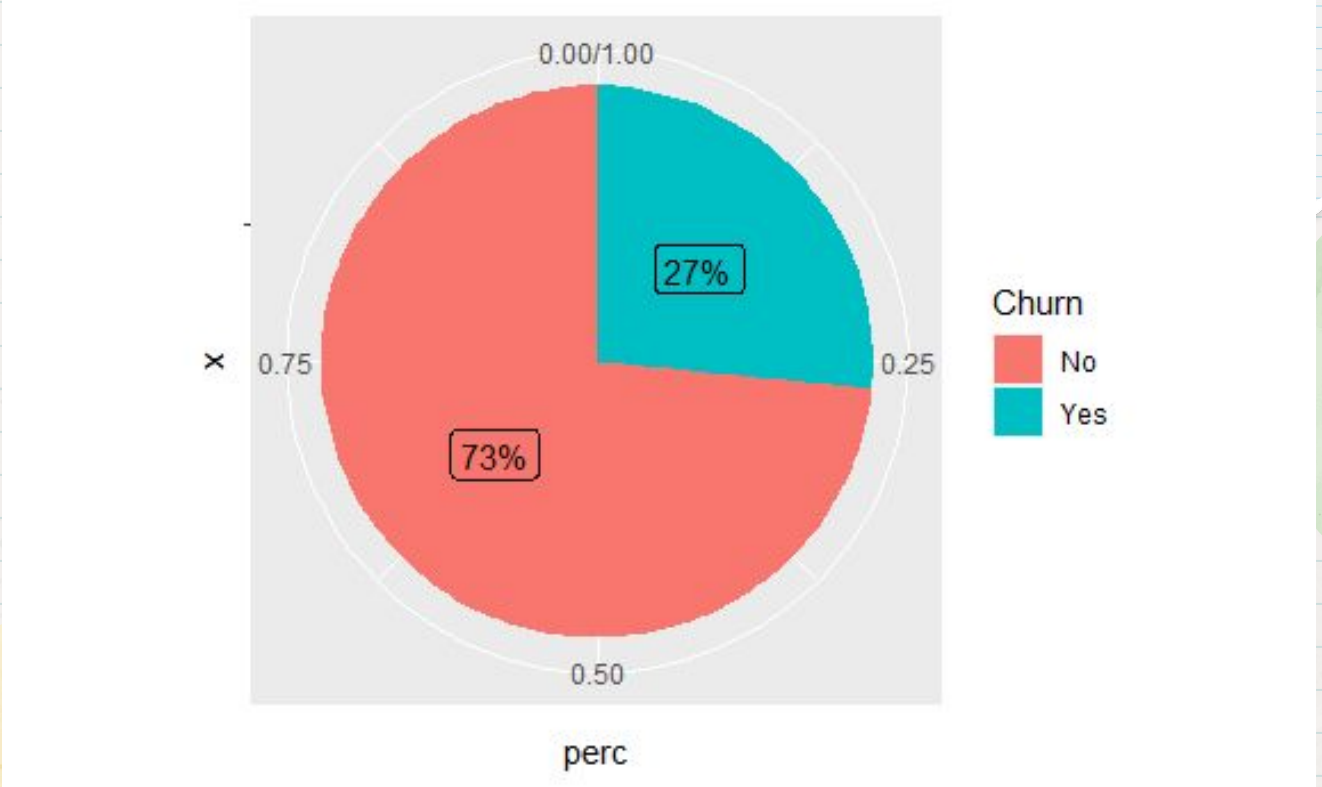
02

變量分析

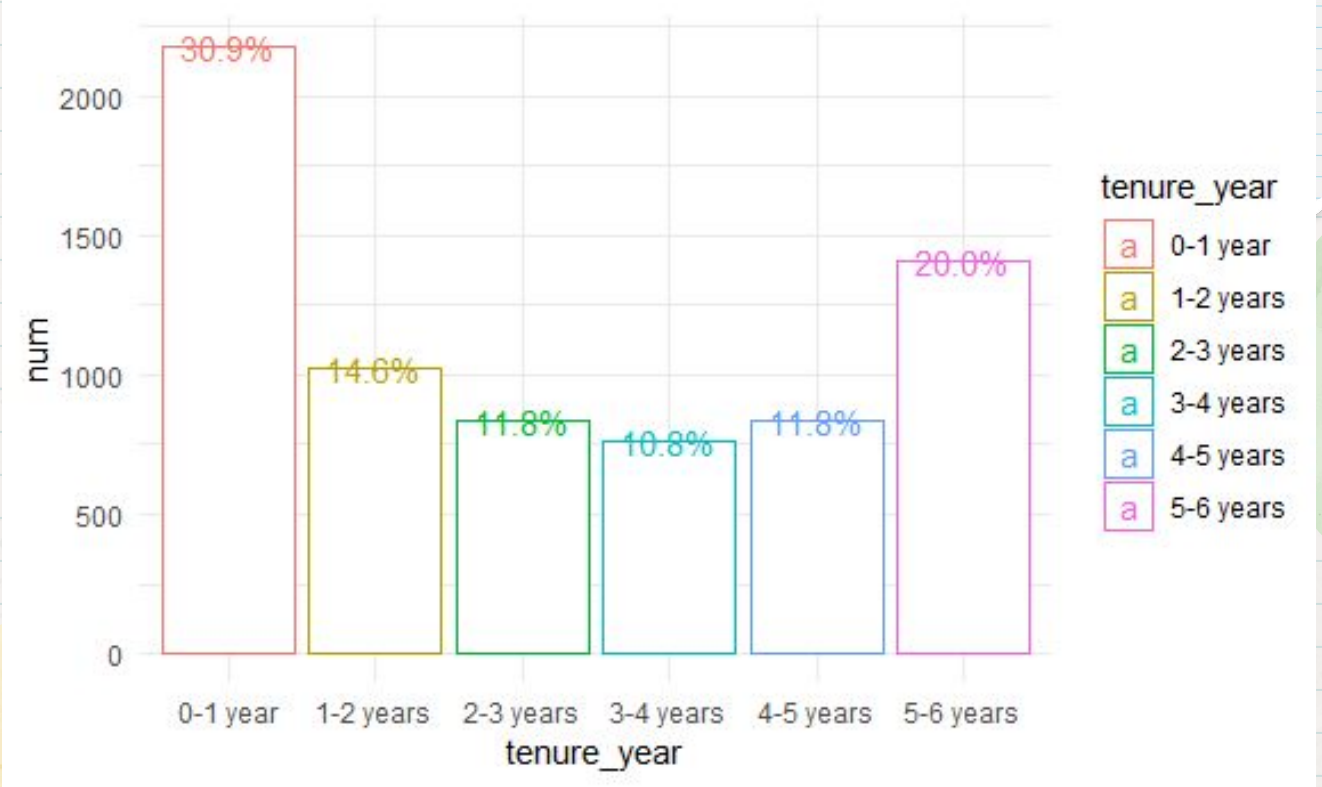
單變量分析-男女比



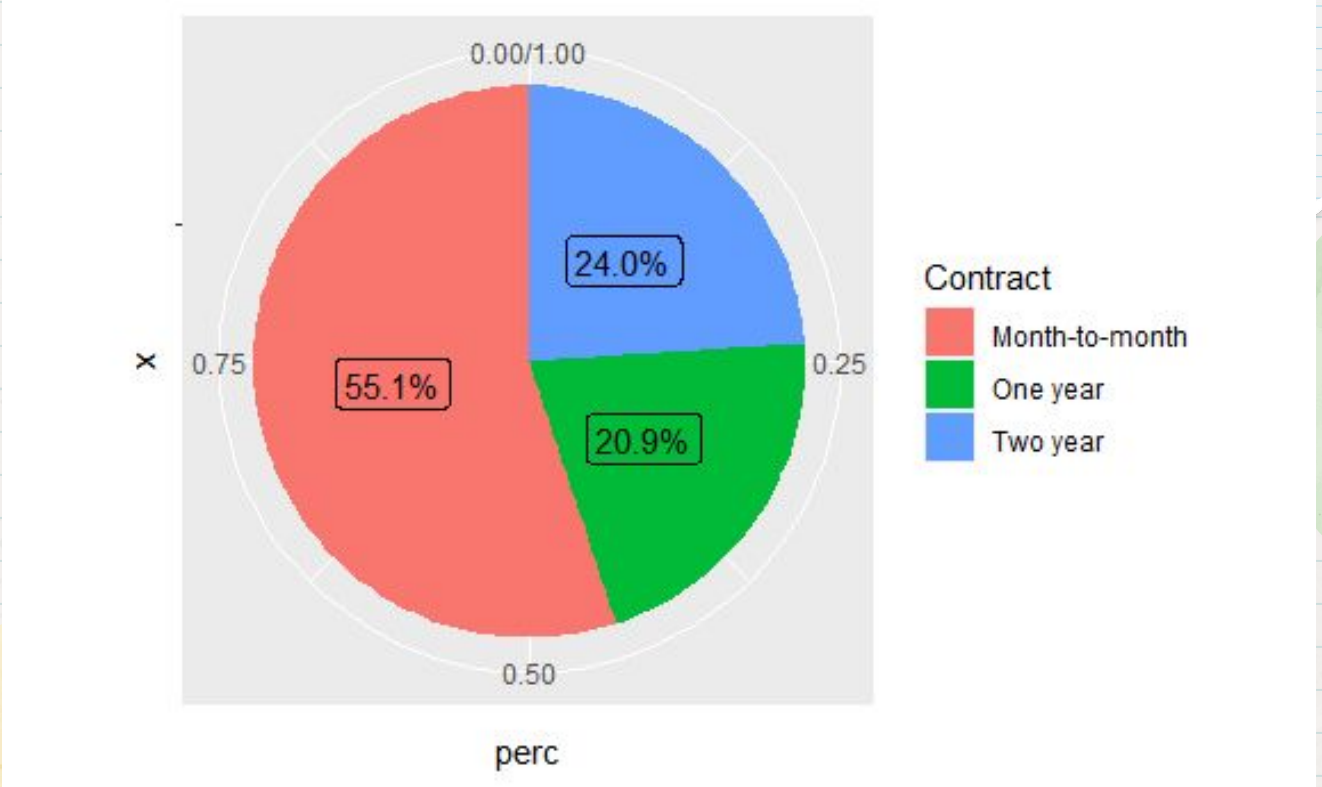
單變量分析-流失率



單變量分析-已簽約期間


30.9%
14.6%
11.8%
10.8%
11.8%
20.0%
2000
1500
1000
500
0
num
0-1 year
1-2 years
2-3 years
3-4 years
4-5 years
5-6 years
tenure_year
tenure_year
a 0-1 year
a 1-2 years
a 2-3 years
a 3-4 years
a 4-5 years
a 5-6 years

單變量分析-合約行式


0.00/1.00
0.25
0.50
0.75
x
perc
24.0%
55.1%
20.9%
Contract
Month-to-month
One year
Two year

單變量分析-合約行式


0.75
0.50
0.25
0.00
perc
Month-to-month
One year
Two year
Contract
Contract
Month-to-month
One year
Two year

雙變量分析

欄位名稱	X-squared	df	p-value
Partner	157.5	1	< 2.2e-16
Dependents	186.32	1	< 2.2e-16
gender	0.47545	1	0.4905
PhoneService	0.87373	1	0.3499
StreamingTV	372.46	2	< 2.2e-16
StreamingMovies	374.27	2	< 2.2e-16
PaperlessBilling	256.87	1	< 2.2e-16
PaymentMethod	645.43	3	< 2.2e-16
SeniorCitizen	158.44	1	< 2.2e-16

anova分析

- tenure

```
             Df  Sum Sq Mean Sq F value Pr(>F)
Churn         1  530982  530982    1008 <2e-16 ***
Residuals  7030 3704983     527
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

TotalCharges

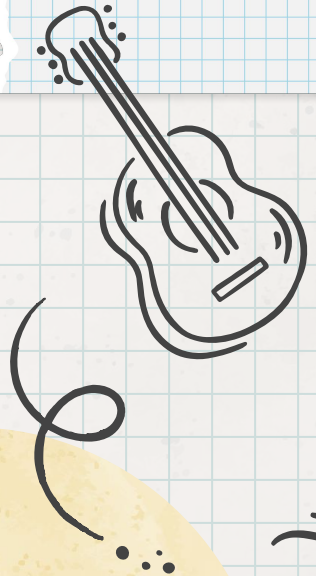
```
             Df    Sum Sq   Mean Sq F value Pr(>F)
Churn         1 1.438e+09 1.438e+09   291.3 <2e-16 ***
Residuals  7030 3.469e+10 4.934e+06
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Sampling

	Yes	No
table(train$Churn)原始	1317	3605
table(train$Churn) OverSampling	3605	3605
table(train$Churn) downsample	1317	1317

使用forward selection選擇變數

選擇的變數		
Contract	InternetService	tenure
PaymentMethod	MultipleLines	OnlineSecurity
TotalCharges	TechSupport	PaperlessBilling
StreamingTV	SeniorCitizen	Partner
StreamingMovies	MonthlyCharges	



03

Random Forest

模型

```
rf.cv <- train(f , data = train,
               method = 'rf',
               metric = 'Accuracy',
               trControl = trControl,
               tuneGrid = tunegrid,
               ntree = 100,
               nodesize = 75)
```

混淆矩陣–原始

Training Evaluation		reference	
		yes	no
prediction	yes	734	292
	no	583	3313

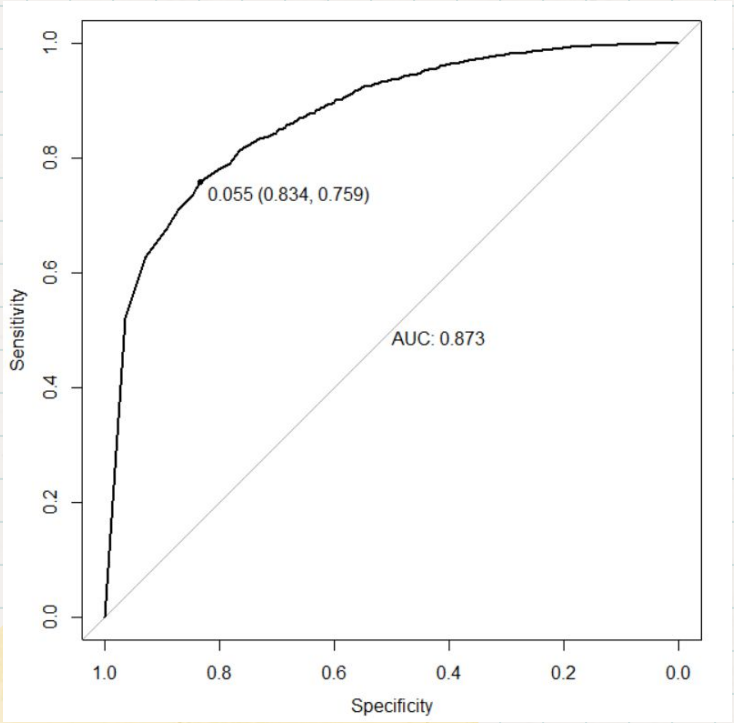
Accuracy	0.8222
Sensitivity	0.5573
Specificity	0.9190

Testing Evaluation		reference	
		yes	no
prediction	yes	295	153
	no	257	1405

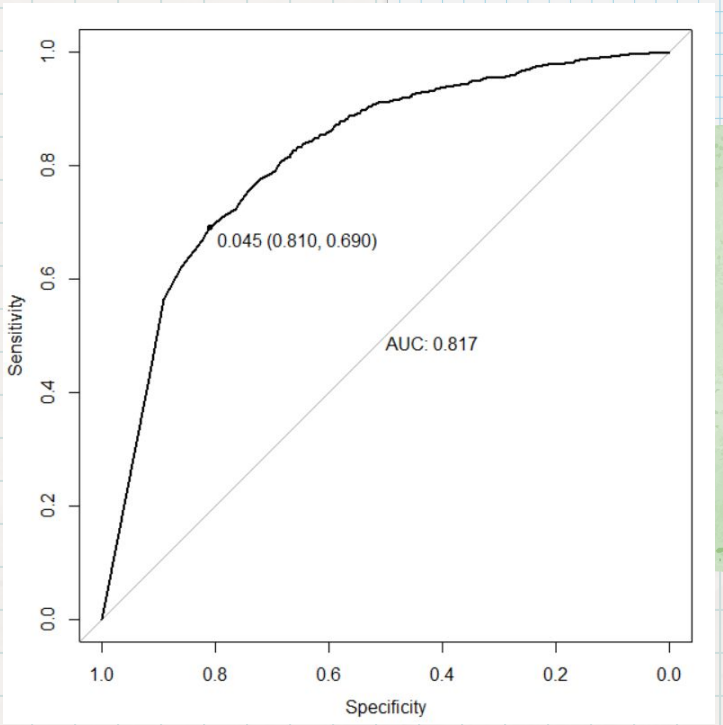
Accuracy	0.8057
Sensitivity	0.5344
Specificity	0.9017

auc

Training Evaluation



Test Evaluation



混淆矩陣-OverSampling

Training Evaluation		reference	
		yes	no
prediction	yes	3097	776
	no	508	2829

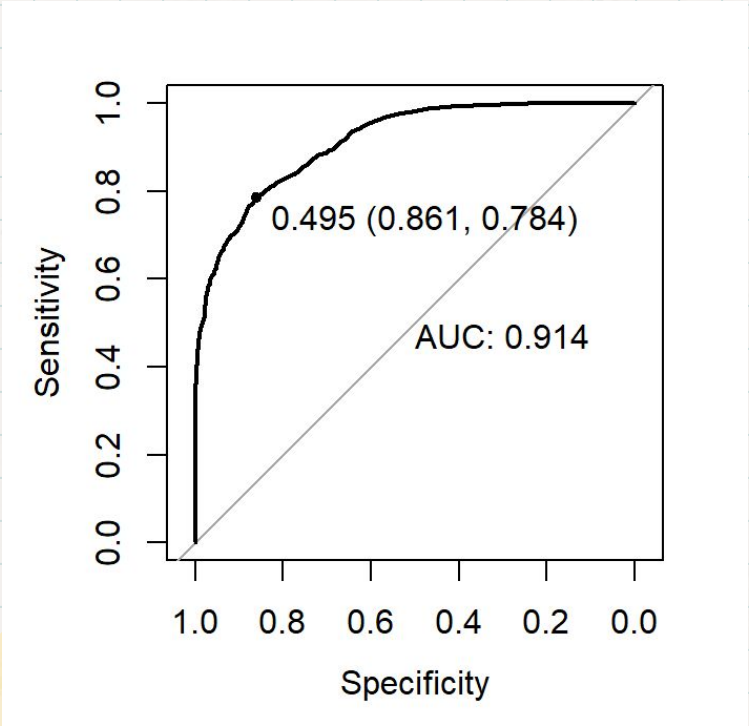
Accuracy	0.8219
Sensitivity	0.8591
Specificity	0.7847

Testing Evaluation		reference	
		yes	no
prediction	yes	429	391
	no	123	1167

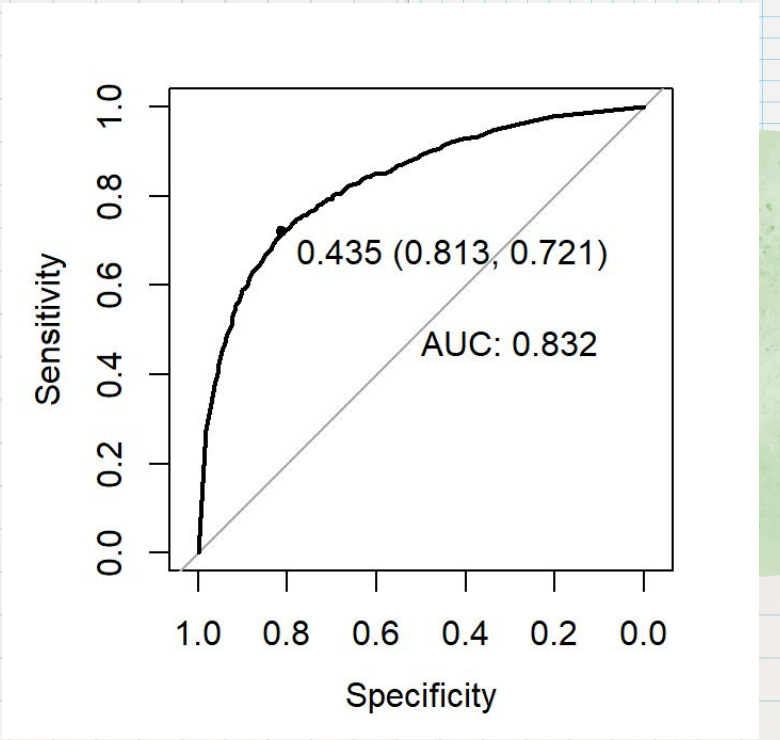
Accuracy	0.8057
Sensitivity	0.7771
Specificity	0.7490

auc-OverSampling

Training Evaluation



Test Evaluation



混淆矩陣-UnderSampling

Training Evaluation		reference	
		yes	no
prediction	yes	1055	290
	no	262	1027

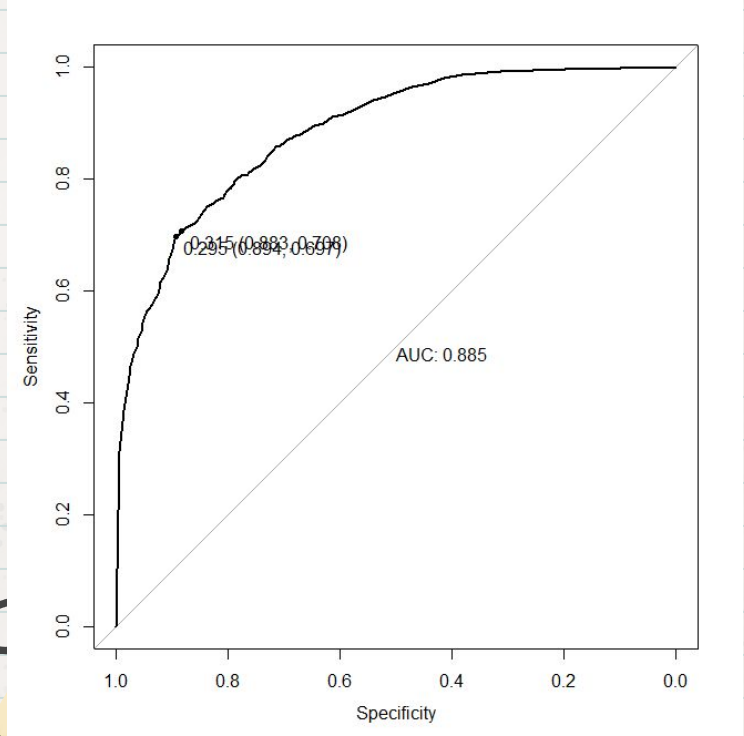
Accuracy	0.7904
Sensitivity	0.8011
Specificity	0.7798

Testing Evaluation		reference	
		yes	no
prediction	yes	433	410
	no	119	1148

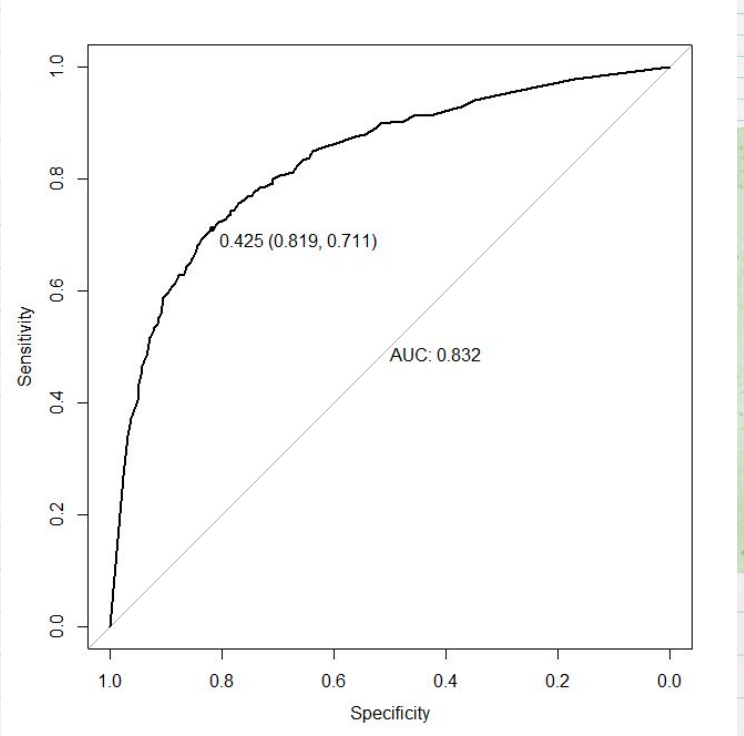
Accuracy	0.7493
Sensitivity	0.7844
Specificity	0.7368

auc-UnderSampling

Training Evaluation



Test Evaluation



04

xgboost

模型

```
tuneGridXGB <- expand.grid(
  nrounds=c(200),
  max_depth = c(2, 3, 4),
  eta = c(0.02),
  gamma = c(1),
  colsample_bytree = c(0.7, 0.8, 0.9),
  subsample = c(0.80),
  min_child_weight = c(1))
```

混淆矩陣

Training Evaluation		reference	
		yes	no
prediction	yes	713	299
	no	604	3306

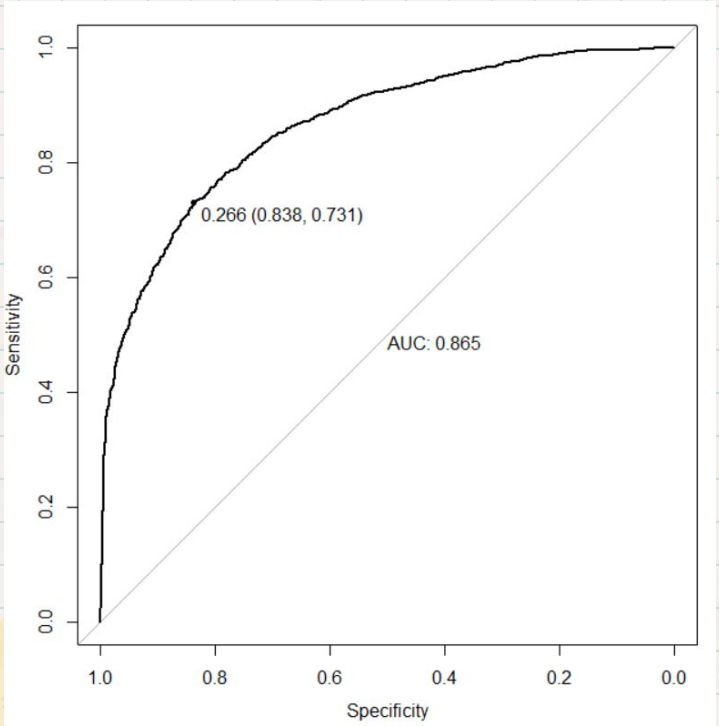
Accuracy	0.8165
Sensitivity	0.5414
Specificity	0.9171

Testing Evaluation		reference	
		yes	no
prediction	yes	297	158
	no	255	1400

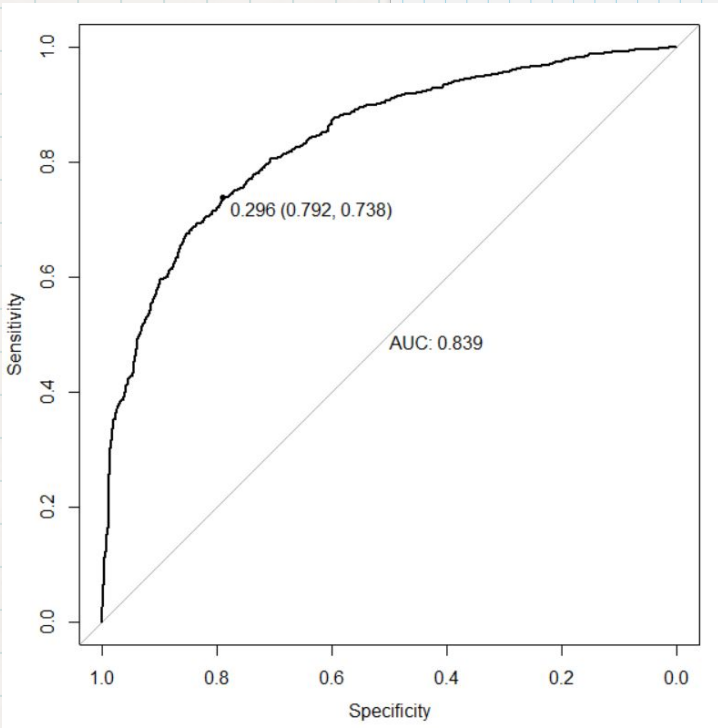
Accuracy	0.8043
Sensitivity	0.5380
Specificity	0.9223

auc

Training Evaluation



Test Evaluation



混淆矩陣-OverSampling

Training Evaluation		reference	
		yes	no
prediction	yes	2981	871
	no	624	2734

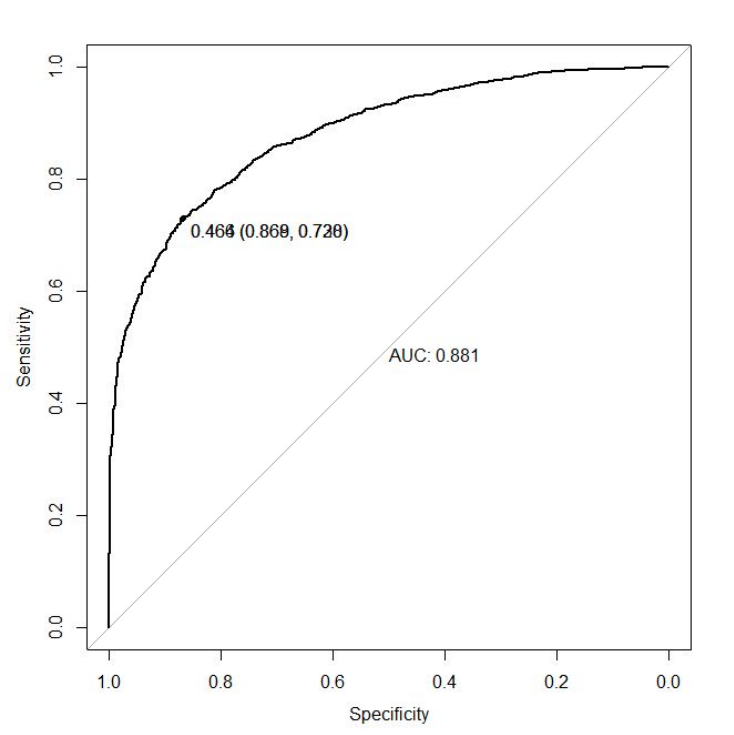
Accuracy	0.7926
Sensitivity	0.8269
Specificity	0.7584

Testing Evaluation		reference	
		yes	no
prediction	yes	437	417
	no	115	1141

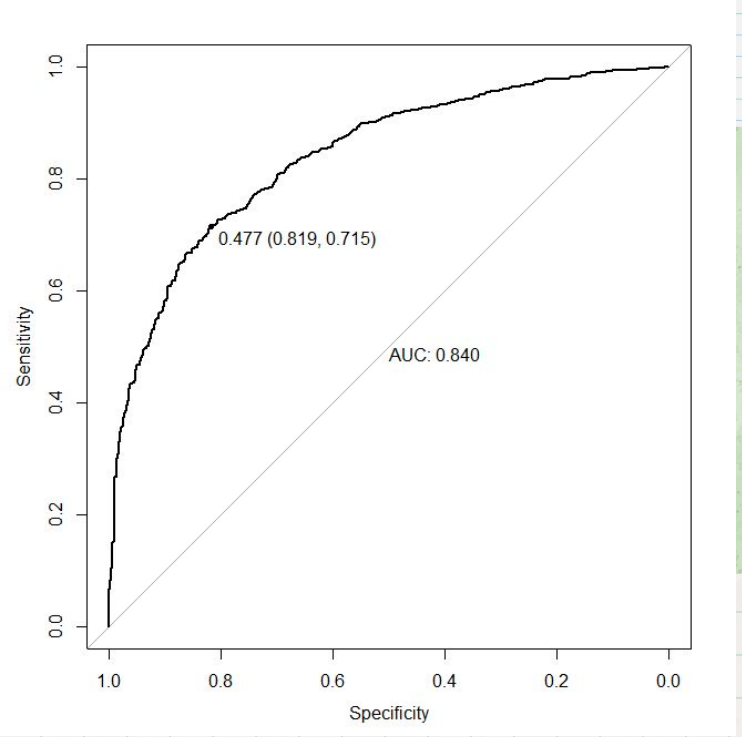
Accuracy	0.7479
Sensitivity	0.7917
Specificity	0.7323

auc-OverSampling

Training Evaluation



Test Evaluation



混淆矩陣-UnderSampling

Training Evaluation		reference	
		yes	no
prediction	yes	1085	336
	no	232	981

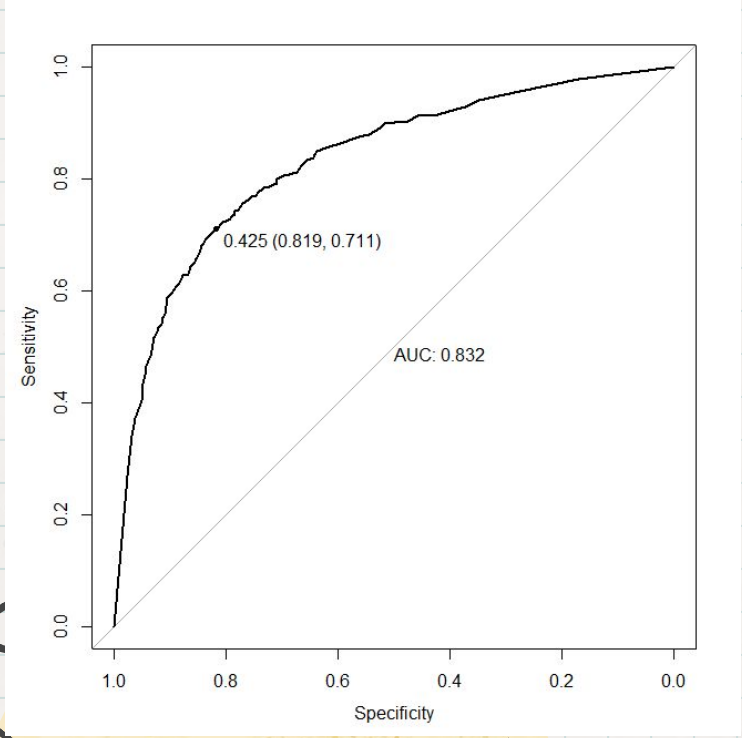
Accuracy	0.7844
Sensitivity	0.8238
Specificity	0.7449

Testing Evaluation		reference	
		yes	no
prediction	yes	448	456
	no	104	1102

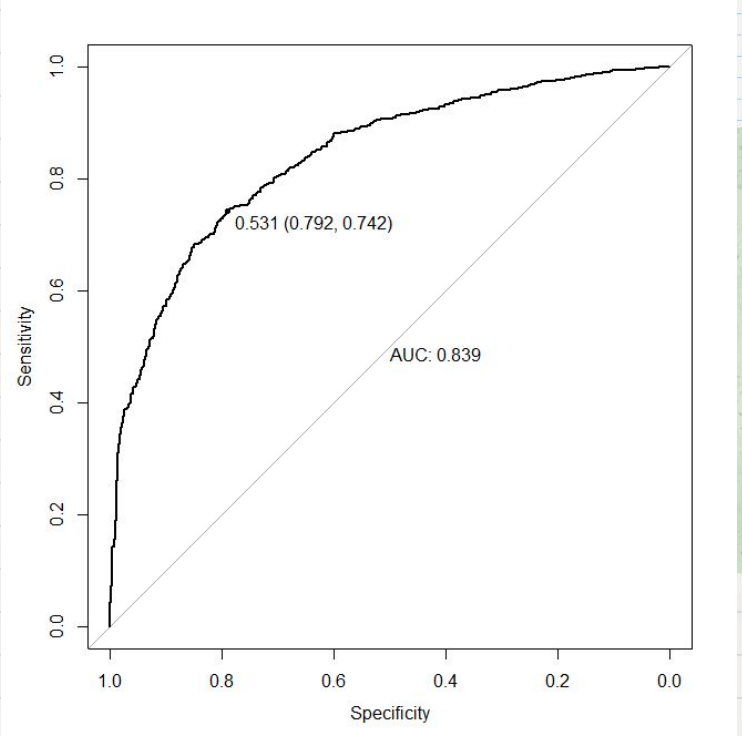
Accuracy	0.7346
Sensitivity	0.8116
Specificity	0.7073

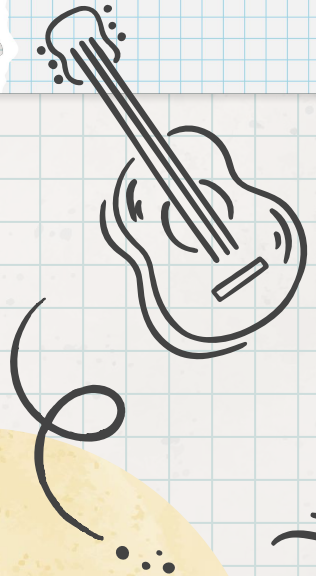
auc-UnderSampling

Training Evaluation



Test Evaluation





05

Logistic Regression

混淆矩陣

Training Evaluation		reference	
		yes	no
prediction	yes	731	386
	no	586	3219

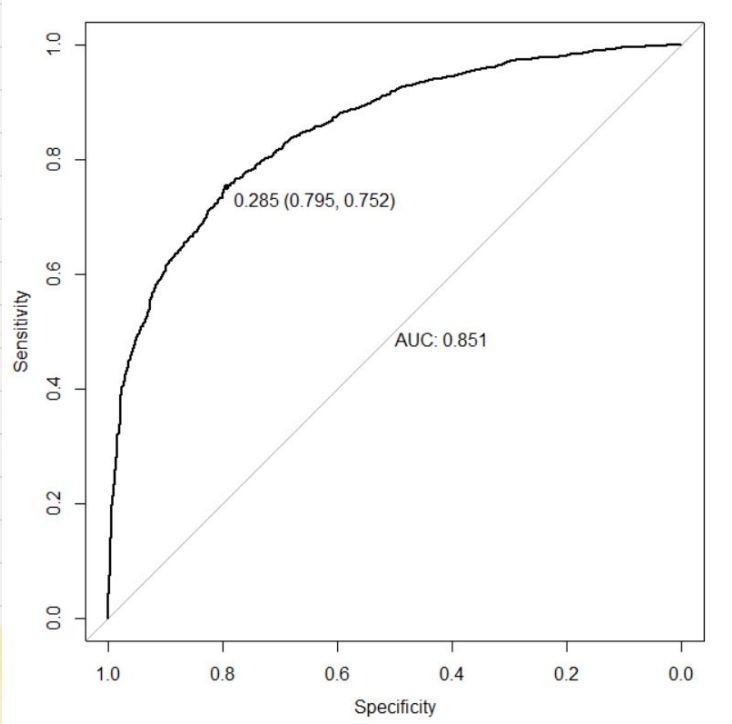
Accuracy	0.8025
Sensitivity	0.5550
Specificity	0.8929

Testing Evaluation		reference	
		yes	no
prediction	yes	315	182
	no	231	1376

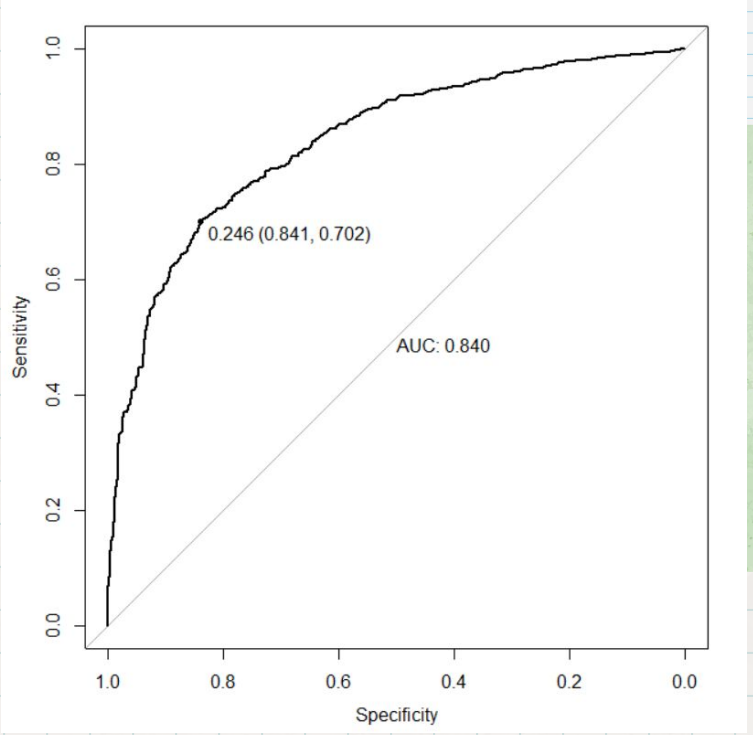
Accuracy	0.8014
Sensitivity	0.5770
Specificity	0.8832

auc-原始

Training Evaluation



Test Evaluation



混淆矩陣-OverSampling

Training Evaluation		reference	
		yes	no
prediction	yes	2909	983
	no	696	2622

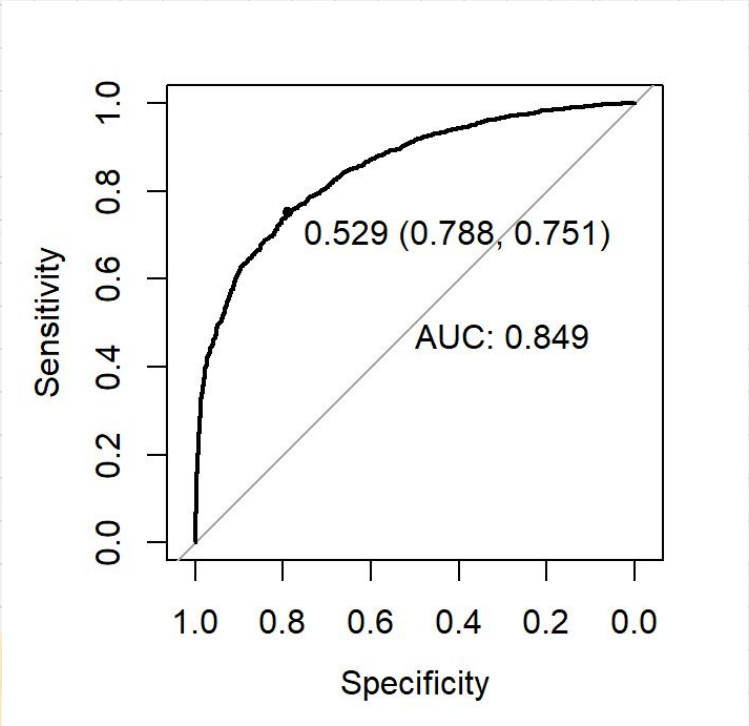
Accuracy	0.7671
Sensitivity	0.8069
Specificity	0.7273

Testing Evaluation		reference	
		yes	no
prediction	yes	449	456
	no	103	1102

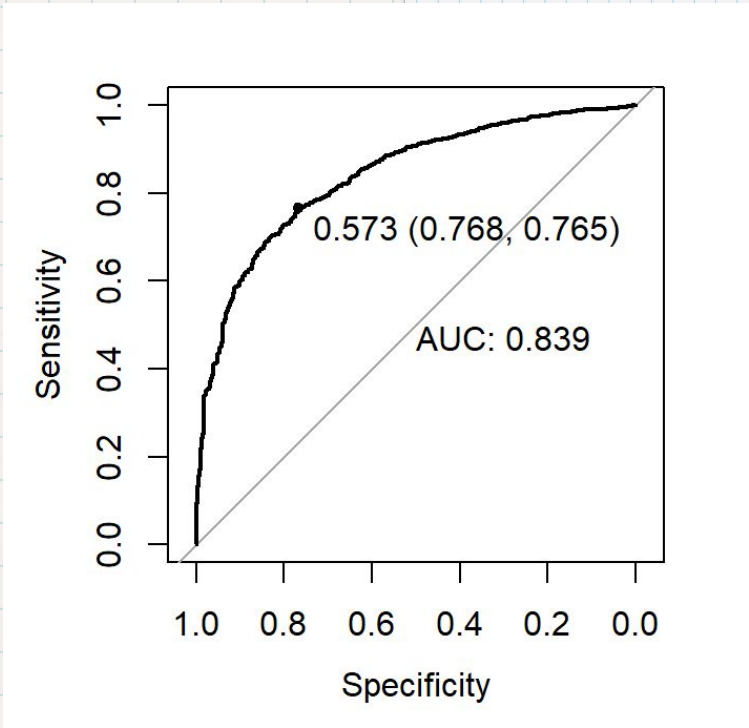
Accuracy	0.7351
Sensitivity	0.8134
Specificity	0.7073

auc-OverSampling

Training Evaluation



Test Evaluation



混淆矩陣-UnderSampling

Training Evaluation		reference	
		yes	no
prediction	yes	1056	360
	no	261	957

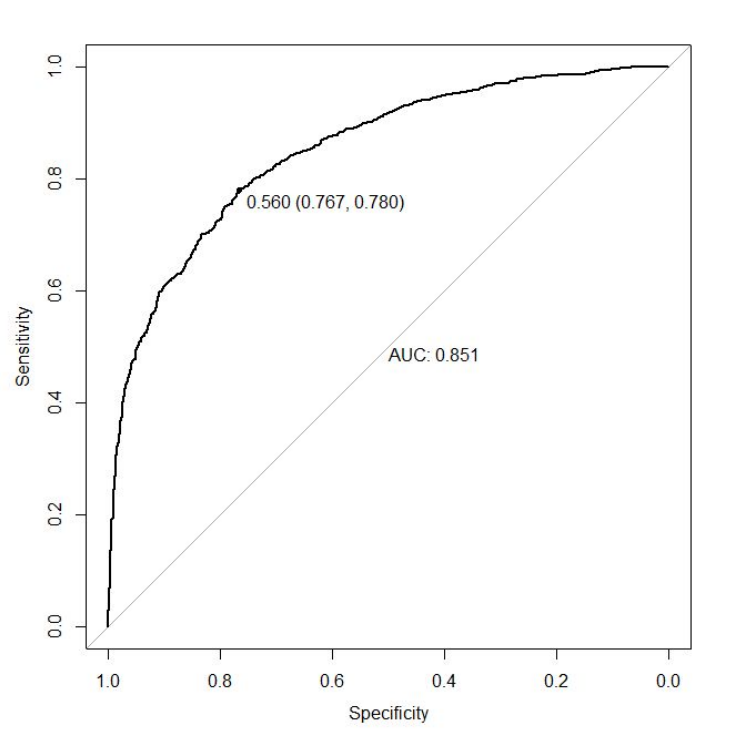
Accuracy	0.7642
Sensitivity	0.8018
Specificity	0.7267

Testing Evaluation		reference	
		yes	no
prediction	yes	444	450
	no	108	1108

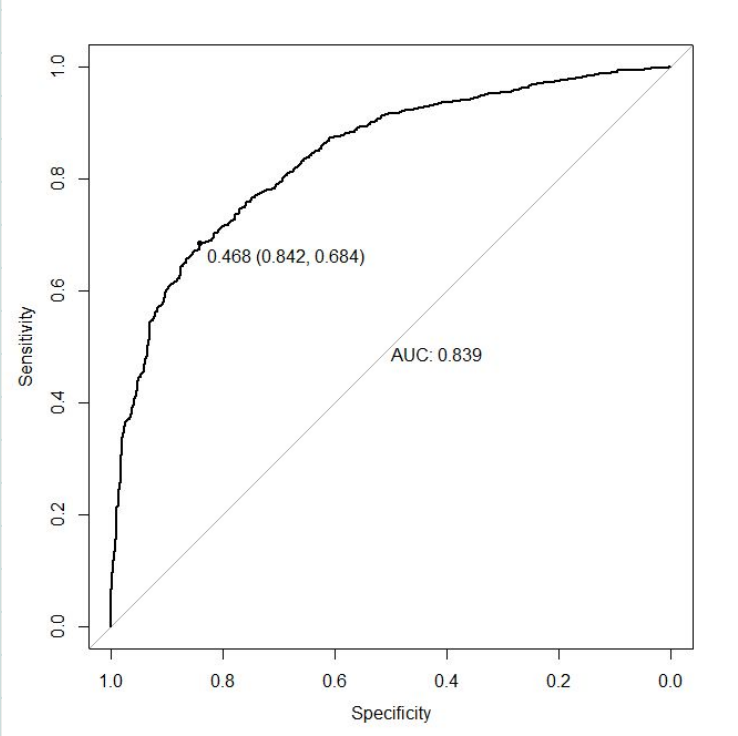
Accuracy	0.7355
Sensitivity	0.8043
Specificity	0.7112

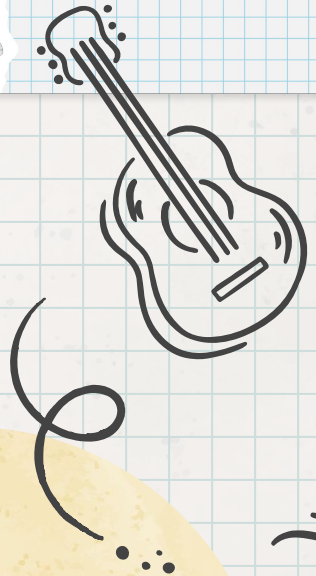
auc-UnderSampling

Training Evaluation



Test Evaluation





06

neural network

混淆矩陣–原始

Training Evaluation		reference	
		yes	no
prediction	yes	698	325
	no	619	3280

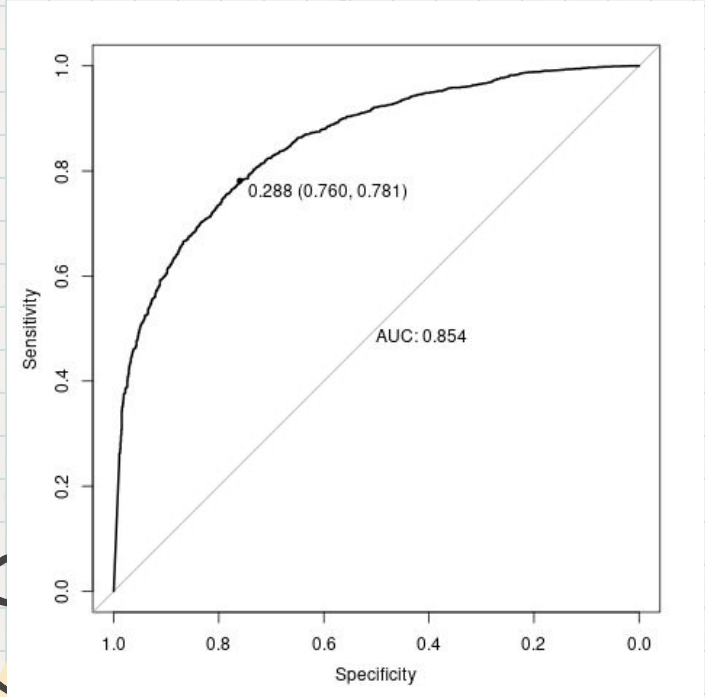
Accuracy	0.8082
Sensitivity	0.5300
Specificity	0.9098

Testing Evaluation		reference	
		yes	no
prediction	yes	301	159
	no	251	1399

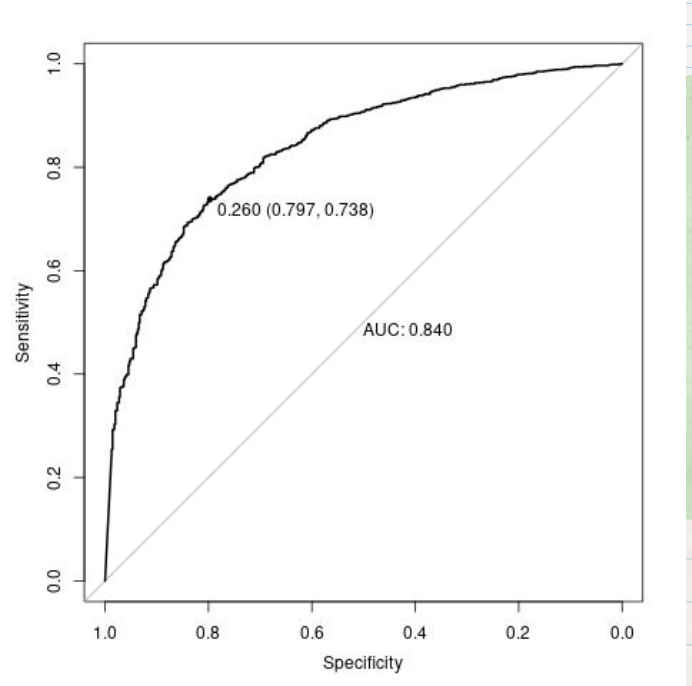
Accuracy	0.8057
Sensitivity	0.5453
Specificity	0.8979

auc-原始

Training Evaluation



Test Evaluation



混淆矩陣-OverSampling

Training Evaluation		reference	
		yes	no
prediction	yes	2816	845
	no	789	2760

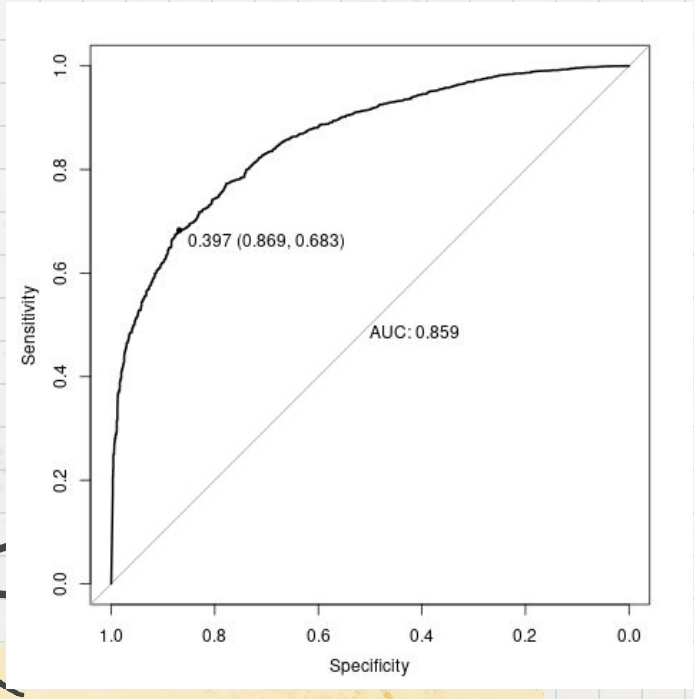
Accuracy	0.7734
Sensitivity	0.7811
Specificity	0.7656

Testing Evaluation		reference	
		yes	no
prediction	yes	422	385
	no	130	1173

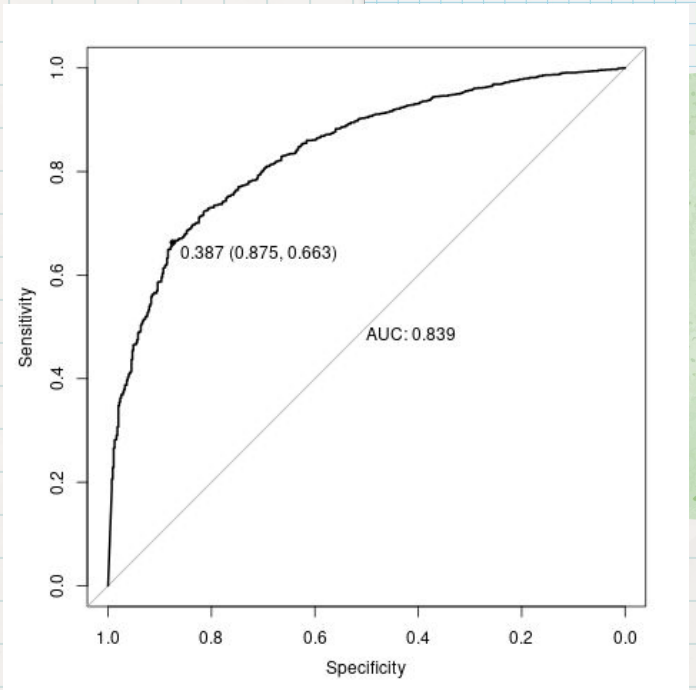
Accuracy	0.7559
Sensitivity	0.7645
Specificity	0.7529

auc-OverSampling

Training Evaluation



Test Evaluation



混淆矩陣-UnderSampling

Training Evaluation		reference	
		yes	no
prediction	yes	1057	366
	no	260	951

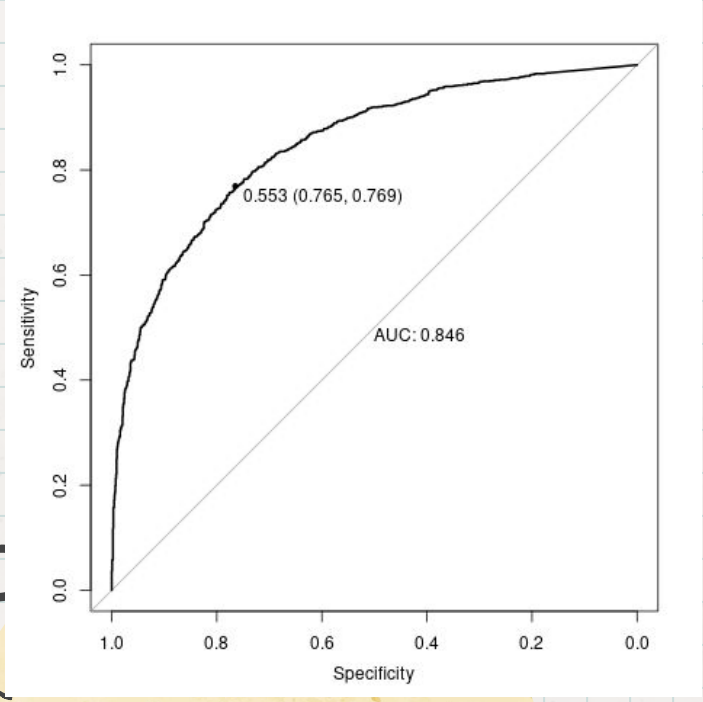
Accuracy	0.7623
Sensitivity	0.8026
Specificity	0.7221

Testing Evaluation		reference	
		yes	no
prediction	yes	444	455
	no	108	1103

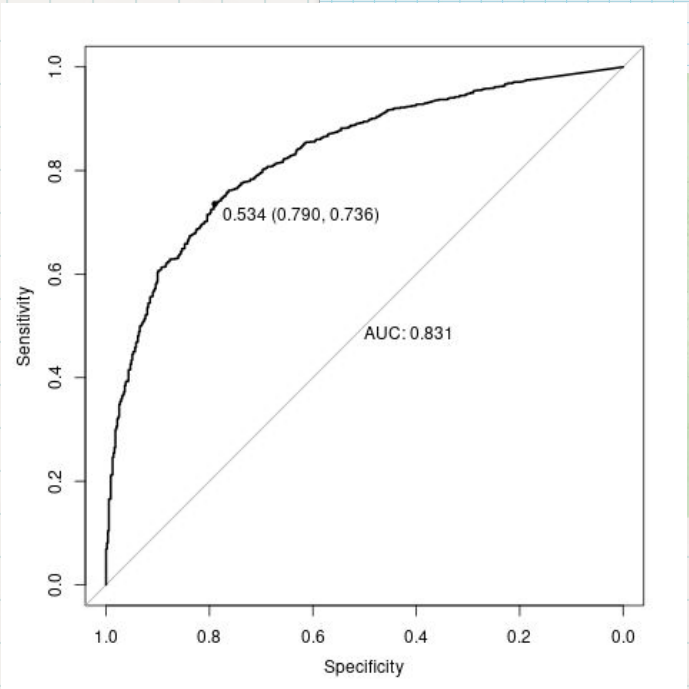
Accuracy	0.7332
Sensitivity	0.8043
Specificity	0.7080

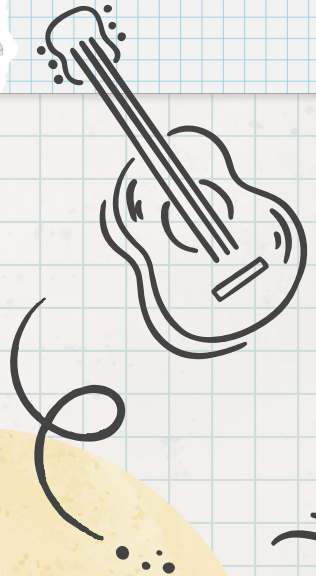
auc-UnderSampling

Training Evaluation



Test Evaluation





07

分析與總結

比較模型–原始資料

	AUC	Accuracy	Sensitivity	Specificity
Random Forest	0.817	0.805	0.534	0.901
xgboost	0.839	0.804	0.538	0.922
Logistic Regression	0.840	0.801	0.570	0.883
neural network	0.854	0.805	0.545	0.897

比較模型-OverSampling

	AUC	Accuracy	Sensitivity	Specificity
Random Forest	0.832	0.805	0.777	0.897
xgboost	0.840	0.747	0.791	0.732
Logistic Regression	0.839	0.735	0.813	0.707
neural network	0.839	0.755	0.764	0.752

比較模型-UnderSampling

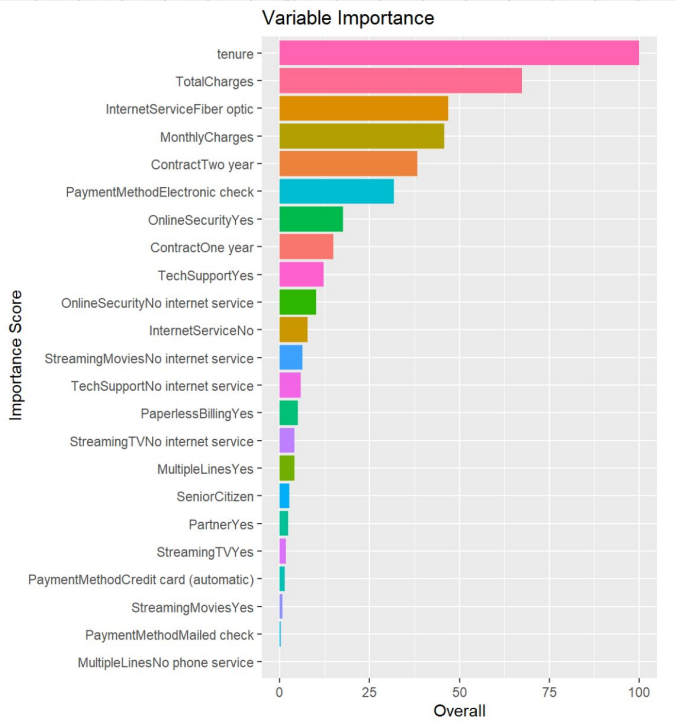
	AUC	Accuracy	Sensitivity	Specificity
Random Forest	0.832	0.749 	0.784	0.736 
xgboost	0.839 	0.734	0.811 	0.707
Logistic Regression	0.839 	0.735	0.804	0.711
neural network	0.839 	0.733	0.804	0.708

解釋-Adjusted OR

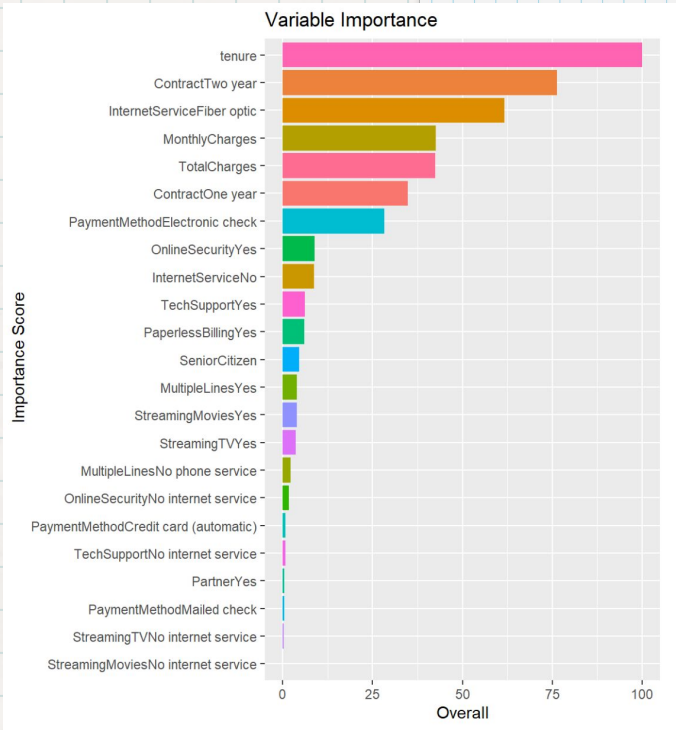
	OR	lower95ci	upper95ci	Pr(>|z|)
`ContractOne year`	1.8816753	1.4653879	2.4162216	7.224837e-07
`ContractTwo year`	4.4149060	2.8904499	6.7433774	6.360509e-12
`InternetServiceFiber optic`	0.2623203	0.1366111	0.5037066	5.818193e-05
InternetServiceNo	3.7019756	1.7623722	7.7762369	5.475217e-04
tenure	4.2324296	2.9463697	6.0798413	5.846884e-15
`PaymentMethodCredit card (automatic)`	1.1424251	0.8725381	1.4957914	3.328638e-01
`PaymentMethodElectronic check`	0.7432355	0.5952335	0.9280374	8.815141e-03
`PaymentMethodMailed check`	1.0369484	0.7927067	1.3564437	7.911902e-01
`MultipleLinesNo phone service`	0.8563892	0.4848422	1.5126622	5.932681e-01
MultipleLinesYes	0.6793757	0.5429173	0.8501318	7.268521e-04
OnlineSecurityYes	1.4400408	1.1410228	1.8174198	2.134082e-03
TotalCharges	0.4955998	0.3385760	0.7254475	3.050034e-04
TechSupportYes	1.3353809	1.0504636	1.6975763	1.817467e-02
PaperlessBillingYes	0.7547954	0.6343523	0.8981067	1.516114e-03
StreamingTVYes	0.6177777	0.4505749	0.8470274	2.780705e-03
SeniorCitizen	0.7743998	0.6377941	0.9402643	9.822628e-03
PartnerYes	1.1512940	0.9767806	1.3569864	9.298738e-02
StreamingMoviesYes	0.6853429	0.4997099	0.9399351	1.906278e-02
MonthlyCharges	1.8973697	0.8951824	4.0215402	9.470953e-02

解釋-Important Matrix

Random Forest

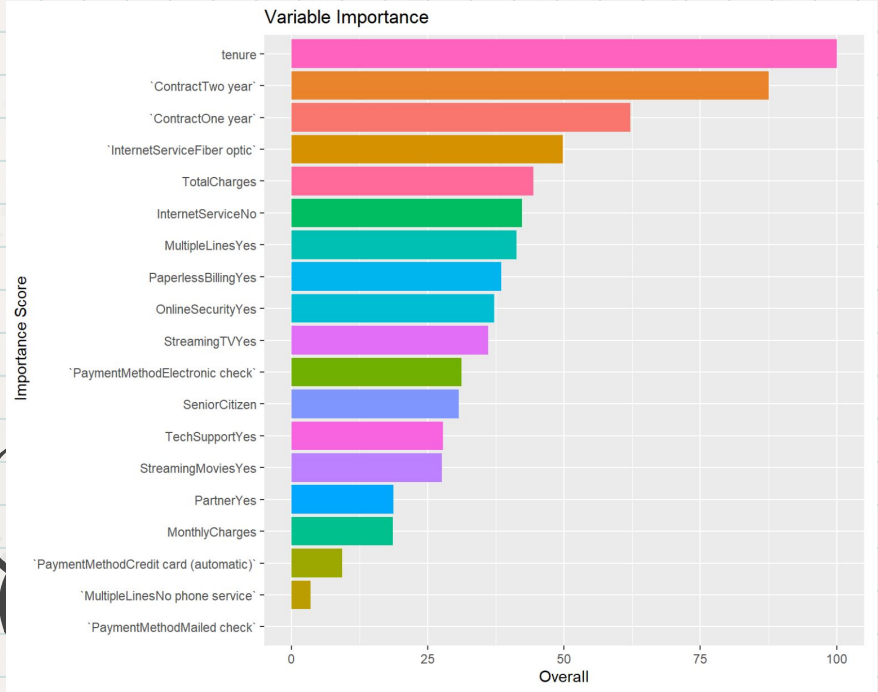


Xgboost

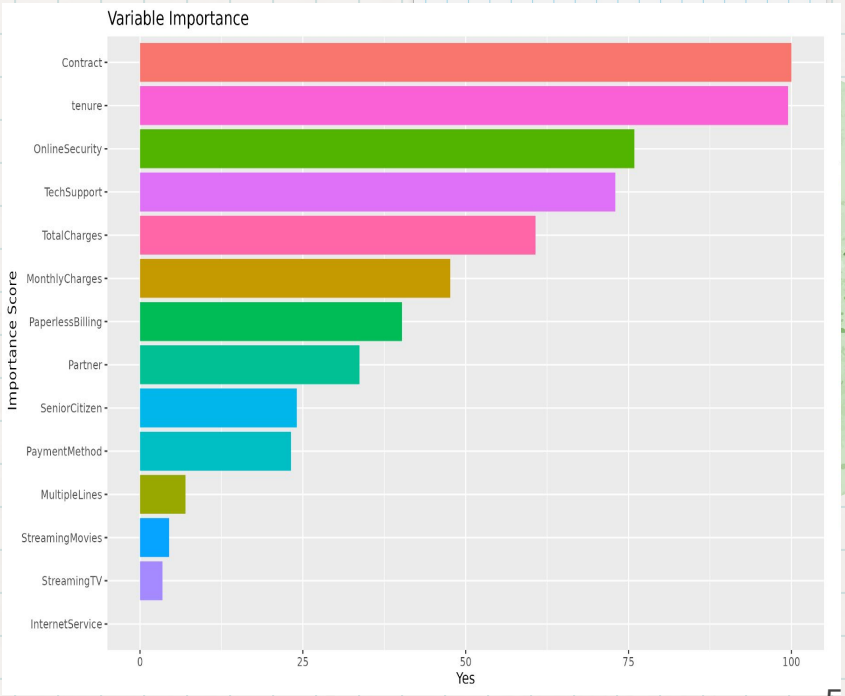


解釋-Important Matrix

Logistic Regression



neural network



結論–原始資料

1.Neural Network擁有最佳的AUC

2.Accuracy其實4種方法差不多，只有些微的差距

3.Logistic Regression有較高的Sensitivity

4.Xgboost有最好的Specificity

結論-OverSampling

1.Xgboost擁有最佳的AUC

2.Accuracy與Specificity以RandomForest最高

3.Logistic Regression有較高的Sensitivity

結論-UnderSampling

1.Xgboost、Logistic Regression與Neural Network有較高的AUC

2.RandomForest有最好Accuracy與Specificity

3.Xgboost有較高的Sensitivity

Thanks!