Mandatory Excercise 2

HeleneEriksen

11/15/2021

Question 1

Split your dataset into a training and a testing set

```
IRS_audits.split <- IRS_audits %>%
  initial_split(prop = 0.8)
train_data <- training(IRS_audits.split ) %>%
drop_na(turnover_volume, total_capital, income_taxes, market_value, prices_adj)
test_data <- testing(IRS_audits.split ) %>%
drop_na(turnover_volume, total_capital, income_taxes, market_value, prices_adj)
```

Question 2 and 3

Fit at least two linear regression models and two logistic regression models, where the second set of models include more predictors than the first. Use IRS_audit as the outcome. Estimate an additional linear regression and a logistic regression but add at least one non-lineartransformation. You do not have to provide an in-depth interpretation of the coefficient.

Linear regression models

```
mod1 <- lm(IRS_audit ~ prices_adj + year + total_capital, data = train_data)
mod2 <- lm(IRS_audit ~ turnover_volume+ total_capital+ income_taxes+ market_value+ prices_adj, data = t
mod1_NonLinear_transformation <- lm(IRS_audit ~ (market_value^3) + (prices_adj^2), data = train_data)
```

logistic regression models

```
m1 <- glm(IRS_audit ~ prices_adj + year + total_capital, data = train_data,
          family = binomial(link = "logit"))
m2 <- glm(IRS_audit ~ turnover_volume+ total_capital+ income_taxes+ market_value+ prices_adj+ year,
          data = train_data,
          family = binomial(link = "logit"))
m3_NonLinear_transformation <-glm(IRS_audit ~ (turnover_volume^2)+ (total_capital^3)+ (income_taxes^4)+
          data = train_data,
          family = binomial(link = "logit"))
```

coefficients for logistic models

```
coef(m1)
```

```
##   (Intercept)    prices_adj          year total_capital
##  3.860411e+01  2.135663e-03 -1.974220e-02  2.174869e-09
```

```
coef(m2)
```

```
##     (Intercept) turnover_volume   total_capital    income_taxes    market_value
##    4.000437e+01    3.057390e-08   -1.858401e-09    3.682368e-08    5.305634e-06
##      prices_adj            year
##    1.948525e-03   -2.048812e-02
```

```
coef(m3_NonLinear_transformation)
```

```
##     (Intercept) turnover_volume   total_capital    income_taxes    market_value
##   -1.050068e+00    2.531212e-08   -1.742178e-09    5.112745e-08    5.702427e-06
```

coefficients for linear models

```
coef(mod1)
```

```
##   (Intercept)    prices_adj          year total_capital
##  8.369778e+00  4.926415e-04 -4.039244e-03  4.927117e-10
```

```
coef(mod2)
```

```
##     (Intercept) turnover_volume   total_capital    income_taxes    market_value
##    2.298207e-01    7.232572e-09   -4.355766e-10    1.024534e-08    1.132699e-06
##      prices_adj
##    4.518190e-04
```

```
coef(mod1_NonLinear_transformation)
```

```
##  (Intercept) market_value   prices_adj
## 2.313663e-01 1.381036e-06 4.364879e-04
```

Discuss the models and interpret your results

It is interesting to look at the coefficients for the different models. We can see the one with the largest coefficients is for the linear model number 2, which is the one with the most amount of parameters. It has a coefficients of 7.23 which means that every time the turnover volume of a company goes up by one, the prediction will have a higher chance of getting an IRS audit.

Another one that is high is the income tax in the logistic model, with non-linear transformations. Here if you have a high income tax, there is a higher change of getting a visit by the IRS, which makes sense, since if they have a high income tax, then they are more likely to be a larger company, and therefore is of more interest to IRS.

Lasso and Ridge regression

Ridge - linear Regression:

```
train_data_Ridge <- train_data %>% drop_na()
outcome <- train_data_Ridge %>% dplyr::select(IRS_audit) %>% as.matrix()
X_vars <- train_data_Ridge %>% dplyr::select(-IRS_audit) %>% as.matrix()
ridge <- cv.glmnet(x = X_vars, y = outcome,
                   alpha = 0,
                   nfolds = 5, intercept = TRUE, family = "binomial",
                   type.measure = "mse") # use mse for linear models
ridge_fit <- glmnet(x =X_vars, y = outcome,
                    alpha = 0,  lambda = ridge$lambda.min,
                    standardize = TRUE,
                    family = "binomial")
```

Lasso logistic regression

```
lasso <- cv.glmnet(x = X_vars, y = outcome,
                   family = "binomial", alpha = 1, nfolds = 5,
                   parallel = TRUE, intercept = TRUE,
                   type.measure = "class") #Use for logistic model
```

```
## Warning: executing %dopar% sequentially: no parallel backend registered
```

```
lasso_fit <- glmnet(x = X_vars, y = outcome, alpha = 1,  lambda = lasso$lambda.min,
                    standardize = TRUE,
                    family = "binomial")
```

These are the top predictors for lasso and ridge

```
df
```

```
##       lasso_est            lasso_choice    ridge_est      ridge_choice
## 1  1.663953e-04              prices_adj 5.498203e-04        prices_adj
## 2  5.694210e-08            income_taxes 4.864509e-07         employees
## 3  6.700230e-09           pretax_income 2.291190e-07      market_value
## 4  3.686020e-09           total_capital 3.856466e-08      income_taxes
## 5  0.000000e+00                    ...1 1.189240e-08     pretax_income
## 6  0.000000e+00                    year 9.701283e-09   turnover_volume
## 7  0.000000e+00            market_value 3.327053e-09      gross_income
## 8  0.000000e+00            gross_income 1.891820e-09 net_sales_revenue
## 9  0.000000e+00       net_sales_revenue 1.591860e-09     total_capital
## 10 0.000000e+00 operating_profit_margin 1.389936e-09 crnt_total_assets
```

It is interesting to see here that the top predictor for both models are price_adj, which is can see makes sense. If the price of the firms stock has changed a lot, then it would be worth for the IRS to double check that they have done their calculations correctly and have paid the right amount of tax. Especially if there has been a large decrease/increase over a short amount of time. It is interesting to see, that in the other linear models, the price_adj is also one of the high predictors, in all 3 models, where in the logistic models, the price_adj, is not that high of an predictor. i

Question 4

Compare the models using customary metrics and discuss the results. Which model performs betterand why? How does performance vary between the training and testing set? Discuss the reason for thevariation in performance.

RMSE in and out of sample -> Linear Regression

RMSE in and out of sample Logistic regression

RMSE for Linear regression

```
df_lin
```

```
##           names RMSE_mod_1 RMSE_mod_2 RMSE_mod_NL
## 1 out of sample  0.4453523  0.4397742   0.4402962
## 2      insample  0.4509525  0.4469308   0.4476849
```

RMSe for Logistic regression

```
df_log
```

```
##           names  RMSE_m_1  RMSE_m_2 RMSE_m_NL
## 1 out of sample 1.2420082 1.2738275 0.4402962
## 2      insample 0.4509525 0.4469308 0.4476849
```

Accuracy for the different models

```
dataframe
```

```
##          Type Logistic_mod_1 Logistic_mod_2 Logistic_mod_3 Linear_mod_1
## 1 OutOfsample      0.5965909      0.6306818      0.6306818    0.6193182
## 2    InSample      0.5575221      0.5870206      0.5958702    0.5663717
##   Linear_mod_2 Linear_mod_3 lasso_regression Ridge_regression
## 1    0.6306818    0.6306818        0.5714286        0.5714286
## 2    0.6135693    0.6076696        0.6391753        0.6391753
```

The root mean squared error tells us about the absolute fit of the model to the data. It looks at the predicted value and compare it to the actual value. So we want the model with the lowest RMSE. We can see that in-sample, the logistic and linear regressions actually looks like they have the same RMSE which i am not sure why. Out of sample, it seems like that the best model with the lowest RMSE is linear regression model 2.

If we look at the accuracy of the models it tells a different story. If we first look at in-sample then it is the Lasso/Ridge regressions that have the highest accuracy of 63.91. It is however interesting that they end up with the same accuracy.

Of the other logistic and linear regression it is the linear model 2 which has the highest in sample and out of sample accuracy.

DISCLAIMER:

I am not sure that these calculations are correct, and i think i might have missed something with the dataset. There was a lot of NA values, which i was not so sure what i should do about. I tried to pick the parameters with the least amount of NA values in the models. I would like to get some Feedback on how to handle this situation.

Also i am not sure about the calculations that i made in terms of accuracy and RMSE and i could not really make any sense of any of it! I think something went wrong somewhere so if possible i would love some feedback on this.

My whole code, can be found here: https://github.com/Theqcue/DS_Mandatory_excercise_2

I am also a bit unsure about what is meant by non-linear transformations and if I have done this part correct or not. Because i tried both making Lasso/Ridge, which goes in and transform the model by penalization, or where i went in manually and added non-linear transformation. So i would like some feedback on this as well.

Question 5

Discuss why a firm would be interested in predicting IRS audits. Which performance metric is best suited for evaluating a model with that goal in mind?

It would be very interesting for a firm to know if they are going to recieve an IRS audit, so they can prepare for this, before they know they have to do this. Therefore it would be best for the model to have recall metric, so that it incorps as many as the true positives as possible. It would be better for them to prepare, and then IRS does not come, then for them to not prepare and for the IRS to come anyway.