Machine Learning Logistic Regression

Mark Vazquez

April 19, 2020

Logistic Regression

Data Preprocessing

```
dataset <- read.csv('Social_Network_ads.csv')
dataset <- dataset[,3:5]
library(caTools)
split <- sample.split(dataset$Purchased, SplitRatio = .75)
training_set <- subset(dataset, split == T)
test_set <- subset(dataset, split == F)
library(xtable)
print(xtable(dataset[sample(nrow(dataset), 15),]), type = "latex", scalebox='1')
```

% latex table generated in R 4.0.0 by xtable 1.8-4 package % Mon May 04 05:25:11 2020

	Age	EstimatedSalary	Purchased
17	47	25000	1
226	37	53000	0
96	35	44000	0
227	36	126000	1
370	54	26000	1
290	37	78000	1
267	40	75000	0
318	35	55000	0
146	24	89000	0
178	25	22000	0
347	53	72000	1
28	47	30000	1
132	33	31000	0
24	45	22000	1
150	20	74000	0

Scaling our Data

```
training_set[,1:2] = scale(training_set[,1:2])
test_set[,1:2] = scale(test_set[,1:2])
```

Fitting Logistic Regression to the Training set

```
logit = glm(formula = Purchased ~.,
                family = binomial,
```

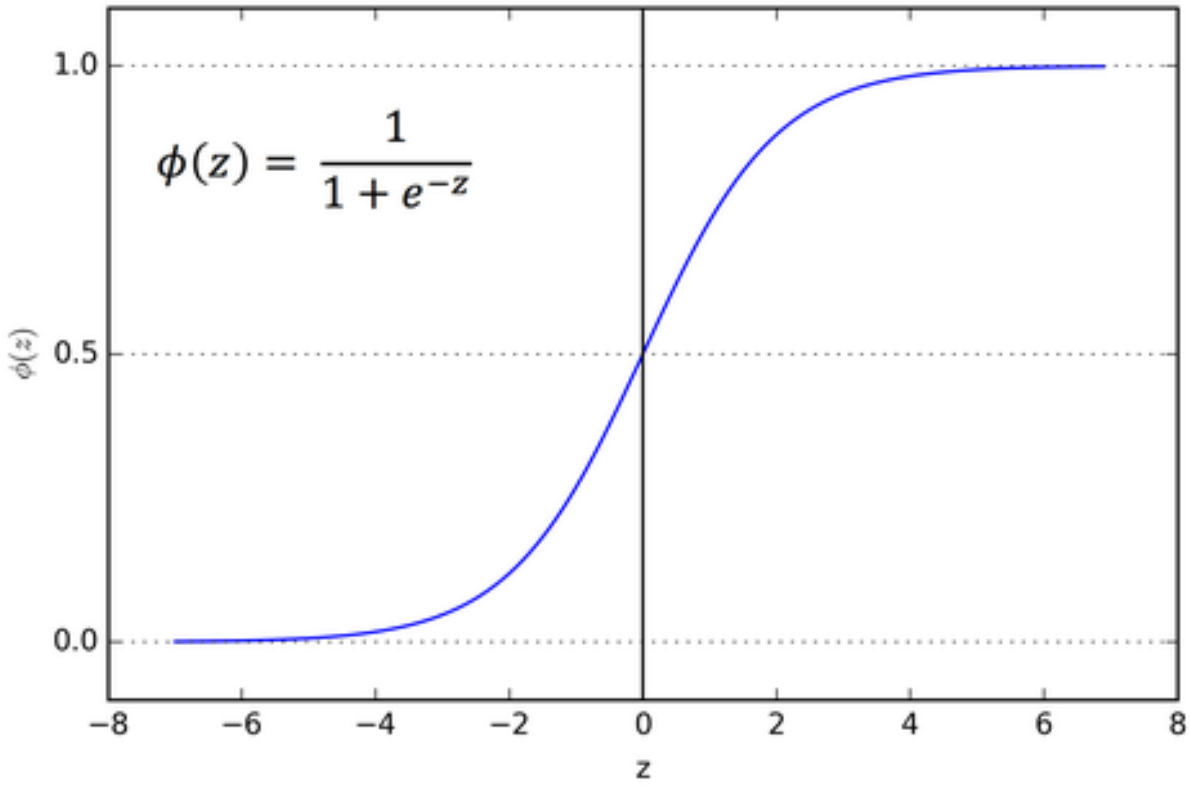


Figure 1: Sigmoid Function

```
               data = training_set)
summary(logit)
```

```
##
## Call:
## glm(formula = Purchased ~ ., family = binomial, data = training_set)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7519  -0.6138  -0.1669   0.4847   2.2356
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -1.0896     0.1861  -5.857 4.72e-09 ***
## Age               2.1329     0.2773   7.691 1.46e-14 ***
## EstimatedSalary   1.1671     0.2007   5.815 6.07e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 390.89  on 299  degrees of freedom
## Residual deviance: 225.71  on 297  degrees of freedom
## AIC: 231.71
##
## Number of Fisher Scoring iterations: 6
```

$$\phi(z) = \frac{1}{1 + e^{-z}}$$

Predicting the Test set Results

```
prob_pred <- predict(logit, type = 'response', newdata = test_set[-3])
head(prob_pred) #look and some y hats
```

```
##          3          7         16         18         21         22
## 0.01313934 0.06804235 0.08597436 0.24179063 0.21595318 0.52484659
```

$$\hat{y} = \begin{cases} 1 & \text{if } \phi(z) \geq 0.5 \\ 0 & \text{otherwise} \end{cases}$$

```
y_pred = ifelse(prob_pred > 0.5, 1, 0)
head(y_pred) #converted out y hat to Binary
```

```
##  3  7 16 18 21 22
##  0  0  0  0  0  1
```

Results in data frame

```
results <- data.frame(table(test_set[,3], y_pred))
results
```

```
##   Var1 y_pred Freq
## 1    0      0   61
## 2    1      0    9
```

```
## 3    0      1    3
## 4    1      1   27
```

```
accuracy <- 83/100
paste("Our accuracy is ", accuracy)
```

```
## [1] "Our accuracy is  0.83"
```

Plotting our Decision Boundry

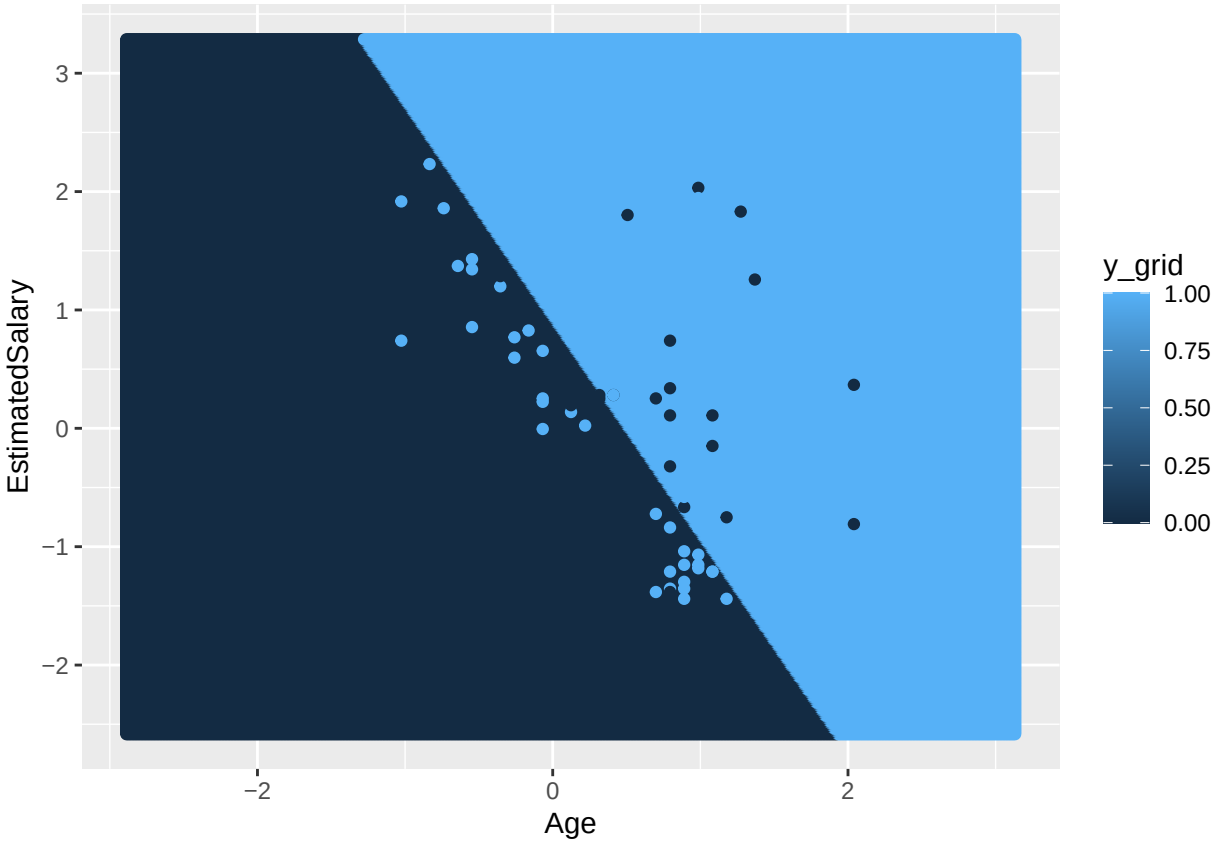
```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------ tidyverse 1.3.0 --
```

```
## v tibble  3.0.1     v dplyr   0.8.5
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## v purrr   0.3.4
```

```
## -- Conflicts --------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
set <- training_set
expand.grid('Age' = seq(min(set[, 1]) - 1, max(set[, 1]) + 1, by = 0.01),
'EstimatedSalary' = seq(min(set[, 2]) - 1, max(set[, 2]) + 1, by = 0.01))%>%
 mutate(prob_set=predict(logit, type = 'response', newdata = .),
 y_grid = ifelse(prob_set > 0.5, 1, 0))%>%
 ggplot()+
 geom_point(aes(x=Age, y=EstimatedSalary, color=y_grid))+
 geom_point(data=training_set, aes(x=Age, y=EstimatedSalary,colour=as.numeric(Purchased)))
```


EstimatedSalary
3
2
1
0
−1
−2
−2
0
2
Age
y_grid
1.00
0.75
0.50
0.25
0.00